Version 1.6

Getting Started

conceptware

GateBuilder © & ^{TM} 2001 by conceptware Alle Rechte vorbehalten.

Das Werk einschließlich aller Teile ist urheberrechtlich geschützt. Dokumentation und Programme sind in der vorliegenden Form Gegenstand eines Lizenzvertrages und dürfen ausschließlich gemäß den in der Packung beiliegenden Vertragsbedingungen verwendet werden. Jede weitergehende Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung in schriftlicher Form von conceptware. Dies gilt insbesondere für Vervielfältigung, Bearbeitung, Übersetzung, Speicherung sowie Bereithaltung zur Nutzung in elektronischen Systemen.

Die in der Dokumentation verwendeten Soft- und Hardwarebezeichnungen und Markennamen der jeweiligen Firmen unterliegen im allgemeinen dem Schutz des Warenzeichen-, Marken- oder Patentrechtes, welches hiermit anerkannt wird.

Die in diesen Unterlagen enthaltenen Angaben und Daten können ohne vorherige Ankündigung geändert werden. Die in den Beispielen verwendeten Firmen, sonstigen Namen und Daten sind frei erfunden, soweit nichts anderes angegeben ist.

Erste Auflage: 23. Mai 2001/ Redaktion: Marckus Kraft

Am Kronberger Hang 2a
65824 Schwalbach a. Ts.

Telefon +49 (0)6196 - 4732 - 0
Fax +49 (0)6196 - 4732 - 115

Internet: www.conceptware.de
Support: support@conceptware.de

Inhaltsverzeichnis

1 Vorwort **1**

1.1 Einleitung . . . 1

1.2 Was ist GateBuilder? . . . 1

1.3 Benutzerspektrum und Handbücher . . . 2

1.4 Voraussetzungen . . . 3

1.5 Schreibweisen . . . 4

2 Workflow Framework **5**

2.1 Aufbau der Workflows . . . 5

2.2 Erweiterung durch eigene Workflows . . . 6

3 Tutorial **7**

3.1 Ziel des Tutorials . . . 7

3.2 Vorgehensweise beim Aufbau einer GateBuilder-Anwendung . . . 8

3.3 Erzeugen einer eigenen Anwendung . . . 9

3.4 Aufbau der WebSite . . . 12

3.5 Konfiguration der Anwendung . . . 17

3.6 Erstellen eigener Content-Typen . . . 22

3.7 Anpassung der application.cfm . . . 30

3.8 Die Aktionen . . . 31

3.9 Relationen für Menü und Kategorienbaum . . . 33

3.10 Veröffentlichen von Inhalten . . . 36

3.11 Erweiterte Aktionen: Hintergrundaktionen für die Bestellung . . . 37

3.12 Toolbars . . . 40

3.13 Design . . . 42

3.14 Ausblick . . . 50

Tabellenverzeichnis **51**

Abbildungsverzeichnis **53**

Index **55**

Kapitel 1

Vorwort

1.1 Einleitung

Herzlich willkommen beim GateBuilder von conceptware, der adaptierbaren Standardsoftware für den schnellen Aufbau dynamischer Web-Portale, Intranets und Extranets.

Als modulare Content- und Community-Management-Plattform wird GateBuilder mit einer Reihe von Anwendungsmodulen ausgeliefert, die *out-of-the-box* einsetzbar und per Web-Browser bedienbar sind. Eingebettet in das offene Framework von GateBuilder können Sie die Anwendungsmodule schnell und sicher auf die Anforderungen Ihrer Organisation maßschneidern.

Als Entwicklungsplattform auf der Basis des Application-Servers Allaire Spectra ermöglicht Ihnen das Framework von GateBuilder des weiteren, Ihre Portal-Lösung mit Hilfe der ColdFusion Markup Langugage (CFML) sehr einfach um leistungsfähige Funktionen und Geschäftsprozesse zu erweitern. Dabei können Sie innerhalb des Frameworks auf mehr als 300 fertig mitgelieferte CustomTags zurückgreifen. Sie stellen direkt nutzbare zentrale Services bereit, beispielsweise für die Bereiche Benutzeroberfläche, Site-Navigation, Personalisierung, Suche, Hilfe, Mehrsprachigkeit, Relationenmanagement und Caching. Darüber hinaus können Sie das Funktionsspektrum ihres GateBuilder-Portals mit Hilfe von tausenden frei oder kommerziell verfügbarer ColdFusion-Bausteine sehr effizient ausbauen. Alle Dienste sind konsequent auf eine komfortable Handhabung ausgelegt. Sie ersparen Ihnen zudem die Auseinandersetzung mit den komplexen Strukturen des zugrunde liegenden Spectra Application-Server.

Wir wünschen Ihnen viel Erfolg bei der Arbeit mit GateBuilder!

1.2 Was ist GateBuilder?

Bei GateBuilder handelt es sich um eine eBusiness Application Suite, mit der auf einfache und komfortable Weise Informationsinhalte unterschiedlichen Typs erzeugt, bearbeitet, verwaltet und im Netz veröffentlicht werden können.

Die Inhalte können ortsunabhängig in Zusammenarbeit von beliebig vielen Anwendern erstellt, sowie zeitnah überarbeitet und auf der WebSite aktualisiert werden.

Zusätzlich läßt sich mit GateBuilder zielgerichtet und flexibel steuern, welchen Benutzergruppen die Information jeweils zur Verfügung steht.

Darüber hinaus können auch die Anwender im Rahmen der sogenannten Personalisierung entscheiden, welche Inhalte und Informationen sie sehen möchten, über welche Neuerscheinungen sie benachrichtigt werden wollen und in welcher Sprache ihnen Inhalte und die Benutzeroberfläche der Anwendung zur Verfügung gestellt werden.

GateBuilder enthält weiterhin erste eBusiness-Module, wie z.B. CalendarGate, ContactGate, DocumentGate, LinkGate, ProductGate, BannerGate und InfoGate.

1.3 Benutzerspektrum und Handbücher

Es werden sechs verschiedene Gruppen von Anwendern unterschieden. Auf Betreiberseite sind dies die Benutzergruppen Autoren (Contributors), Herausgeber (Editors), Designer/Entwickler (Designers) und Administratoren (Administrators), auf Seiten der WebSite-Besucher handelt es sich um Besucher (WebUsers) und die registrierten Benutzer (SiteMembers).

Im Folgenden werden die Hauptcharakteristika der verschiedenen Anwendergruppen sowie die dazugehörigen Standardbenutzer aufgeführt. Zusätzlich werden die Handbücher und Informationsmaterialien aufgelistet, welche für die jeweilige Gruppe ausschlaggebend sind.

1.3.1 Besucher

- WebUsers
  Bei einem WebUser handelt es sich um einen erstmaligen oder unregelmäßigen Besucher der WebSite, der im Unterschied zum SiteMember noch kein registrierter Benutzer ist. WebUsern stehen Inhalte nur eingeschränkt zur Verfügung, das heißt, der Zugriff auf bestimmte Kategorien kann ausgeschlossen oder auf Titel und Kurzbeschreibung bestimmter Inhalte beschränkt sein.
  ⇒ Benutzerhandbuch, Kapitel GateBuilder für den WebUser

- SiteMembers
  Standardbenutzer: SiteMember
  Kennwort: sitemember
  Mitglieder der Gruppe SiteMembers haben sich im Unterschied zum WebUser bereits beim Site-Betreiber registriert. Sie erhalten damit Zugriff auf weiterführende Informationen sowie zusätzliche Dienstleistungen und Serviceangebote. Auch die Personalisierungskomponente steht ihnen jetzt zur Verfügung.
  ⇒ Benutzerhandbuch, Kapitel GateBuilder für den SiteMember

1.3.2 Betreiber

- Autoren (Contributors)
  Standardbenutzer: Contributor
  Kennwort: contributor
  Der Autor erstellt die Inhalte der Anwendung in einem arbeitsschrittgeführten Prozeß. Dieser Prozeß kann auch arbeitsteilig ablaufen, indem Inhalte anderen Mitarbeitern in einem separaten Schritt zur weiteren Bearbeitung zugewiesen werden. Dieser Schritt verwendet die in GateBuilder enthaltene Workflow-Komponente und ermöglicht somit die Weiterleitung und automatische Benachrichtigung der Mitarbeiter. Abschließend wird der Inhalt an den Editor übersandt, der für die Veröffentlichung der Inhalte im Netz verantwortlich ist.
  ⇒ Benutzerhandbuch, Kapitel GateBuilder für den Contributor

- Redakteure (Editors)
  Standardbenutzer: Editor
  Kennwort: editor
  Der Herausgeber erhält die von den Contributoren angelegten Inhalte zur Überprüfung und zur endgültigen Veröffentlichung. Er kann die Inhalte selbständig bearbeiten, Veröffentlichungszeiträume und Kategorisierung festlegen oder die Artikel zurück an die Contributoren zur erneuten Überarbeitung verweisen. Wurde der Inhalt von ihm für korrekt befunden, gibt er ihn frei und publiziert ihn im Web.
  ⇒ Benutzerhandbuch, Kapitel GateBuilder für den Editor

- Designer/Entwickler (Designers)
  Standardbenutzer: Designer
  Kennwort: designer
  Der Designer hat die Aufgabe, sämtliche in der Anwendung enthaltenen Ressourcen, das heißt, sämtliche sprachabhängigen Textbausteine, sowie die Typ-Spezifikationen zu bearbeiten.

Hierfür besitzt er die gleichen Zugriffsrechte innerhalb der Anwendung wie der Administrator. Über die Schaltfläche ( ) hat er - ohne den Umweg über das Administratormenü - schnellen Zugriff auf den Ressource-Editor oder die Typ-Spezifikation, um die Ressource oder das typbezogene Objektlayout zu editieren.

Der Designer gehört dabei sowohl der Gruppe der Administratoren als auch einer eigenen Gruppe namens Designers an.
⇒ GateBuilder Tutorial
⇒ Administratorenhandbuch

- Administratoren (Administrators)
  Standardbenutzer: Administrator
  Kennwort: administrator
  Der Administrator konfiguriert die Anwendung und das Layout hinsichtlich der firmenspezifischen Anforderungen (beispielsweise Menüleisten, Toolbars, Kategorien etc.). Er legt Benutzer an, ordnet sie Gruppen zu und verwaltet allgemein die Zugriffsrechte.
  ⇒ GateBuilder-Tutorial
  ⇒ Administratorenhandbuch

1.4 Voraussetzungen

Um mit diesem Tutorial effektiv arbeiten zu können, sollten Sie bereits Grundkenntnisse in HTML, ColdFusion und Allaire Spectra besitzen.

Folgende Software sollte auf Ihrem Arbeitsplatz-Rechner installiert sein:

- Web-Server (z.B. Microsoft IIS V4.0)
- ColdFusion Server 4.5.1 und SP2 oder höher
- Allaire Spectra 1.5
- ColdFusion Studio 4.0 oder höher
- Microsoft Internet Explorer $>$4.0
- Sybase Adaptive Server Anywhere 6.03 oder 7
- GateBuilder 1.6

Hinweis

Die Angaben im Tutorial setzen eine lokale GateBuilder-Installation voraus. Wenn Sie im Netzwerk oder auf anderen Laufwerken arbeiten, sollten Sie die entsprechenden Pfade verwenden.

⇒ Programmverzeichnis: `c:\gatebuilder\`
⇒ Web-Verzeichnis: `c:\inetpub\wwwroot\`

Die zum Tutorial gehörenden Lösungsdateien finden Sie in den folgenden Verzeichnissen:

- Web-Dateien: `[c:\inetpub\wwwroot]\gatebuilder\tutorial\`
- CustomTags: `[c:\gatebuilder]\customtags\tutorial\`
- Handlers: `[c:\gatebuilder]\handlers\tutgbProduct\`

Die für die Lösung benötigten Objekte sind in der Datenbank mit _tut gekennzeichnet. Sollten bereits Tutorialobjekte (tut. . .) in Ihrer Datenbank vorhanden sein, löschen Sie diese.

Achtung: Sie erstellen und ändern Inhalte und Objekte in Allaire Spectra bzw. GateBuilder, deshalb sollten Sie sicher stellen, daß Sie mit einer Testdatenbank bzw. in einer Testumgebung arbeiten.

1.5 Schreibweisen

In diesem Tutorial werden folgende Schreibweisen für die verschiedenen Funktionsbereiche der Programmoberfläche verwendet:

Menü
Menüpunkte
Dialoge / Dialogfenster
Optionen
Eingabefeld
SCHALTFLÄCHEN
‚Symbol‘

Tabelle 1.1: Schreibweisen

Kapitel 2

Workflow Framework

2.1 Aufbau der Workflows

Für die Abbildung der Arbeitsabläufe nutzt GateBuilder das Workflow- und Task-Modell von Allaire Spectra. Allerdings sollten die Workflows nicht den kompletten Ablauf abbilden, da dies für das tatsächliche Arbeiten zu unflexibel wäre. Zudem soll es möglich sein ein definiertes Workflow Template nicht nur ausschließlich für einen `Artifact Type` zu verwenden, wie dies in Spectra vorgesehen ist, sondern ihn für alle GateTypes zu verwenden.

Name	Label	Beschreibung
01 - Später fortfahren	`gbWorkflow_237`	Erstellt einen Task für die eigene Taskliste
02 - Überarbeiten und Freigeben	`gbWorkflow_197`	Schickt einen Task an einen berechtigten Nutzer (i.d.R. der Editor). Berechtigungen werden über die Security-Relation `gbWorkflowExecute` gesetzt.
03 - Überarbeiten durch Contributor	`gbWorkflow_198`	Schickt einen Task an einen berechtigten Nutzer (i.d.R. einen Contributor)
04 -Veröffentlichen im Web	`gbWorkflow_199`	Veröffentlicht automatisch einen Inhalt im Web (`ATTR_ACTIVE=1`; `ATTR_ARCHIVED=0`)
05 - Ins Archiv stellen	`gbWorkflow_200`	Merkt einen Inhalt für das Archiv vor (`ATTR_ACTIVE=0`; `ATTR_ARCHIVED=1`)

Tabelle 2.1: Liste der Workflow Templates

Die Workflow Templates stellen keine Möglichkeit zur Mehrsprachigkeit zur Verfügung. Um die Bezeichnung der Workflows sprachunabhängig gestalten zu können, wurde folgende Namenskonvention eingeführt:

`gbWorkflow_` + `resource_ID`.

So kann die `resource_ID` für die Bezeichnung aus dem Label übernommen werden.

Jedem GateBuilder-Workflow Template wird derzeit genau ein Task zugeordnet. Das CustomTag `gbWorkflow Initiate` startet aber alle Tasks die einem Workflow Template zugeordnet wurden, es können also auch mehrere Tasks mit einem Workflow verbunden sein.

2.2 Erweiterung durch eigene Workflows

Um eigene Workflows in die GateBuilder-Applikation einzubinden, nehmen Sie folgende Schritte vor:

- Schreiben Sie für alle GateTypes eine gleichnamige Methode, welche vom neuen Task angesprochen wird.
- Erzeugen Sie eine Ressource für die Bezeichnung des neuen Workflows.
- Erzeugen Sie einen Task in Spectra und ordnen Sie die neue Methode zu. Soll der Task automatisch ausgeführt werden, ordnen Sie den Default User SYSTEM zu und setzen das Flag `"BAUTOEXECUTE"`.
- Erzeugen Sie ein Workflow Template mit Namen (`=LABEL`) `gbWorkflow_[resource_ID]` und ordnen Sie diesem den neuen Task zu.
- Definieren Sie in GateBuilder, welche Gruppen das Recht erhalten, diesen Workflow zu initiieren, und welche Gruppen den Workflow ausführen dürfen.

Kapitel 3

Tutorial

Hinweis
Sie erstellen und ändern Inhalte und Objekte in Allaire Spectra bzw. GateBuilder.

Stellen Sie deshalb sicher, daß Sie mit einer Testdatenbank bzw. in einer Testumgebung arbeiten. Vermeiden Sie direkt auf einem Produktivsystem zu arbeiten.

3.1 Ziel des Tutorials

Das Tutorial soll einen schnellen Einstieg in die Entwicklung mit GateBuilder und eine Einschätzung des Aufwands einer Umstellung von Web-Anwendungen auf GateBuilder vermitteln.

Dazu wird beispielhaft, in Bezug auf Layout und Funktionalität jedoch eine vollständige GateBuilder-Anwendung erstellt. Das folgende Szenario soll als Grundlage für die ersten Entwicklungsschritte dienen, um die Grundfunktionen und die Architektur von GateBuilder näher kennenzulernen.

3.1.1 Beispiel: Versandhandel Multipart AG

Für den Versandhandel Multipart AG soll eine GateBuilder-Anwendung erstellt werden. Folgende Rahmenbedingungen werden vorausgesetzt:

Die Multipart AG möchte die einzelnen Produktgruppen direkt auf ihrer Homepage präsentieren. Die Produktgruppen sollen alle Produkte hierarchisch einteilen (z.B. Haushalt → Küche → Zubehör). Jeder Benutzer, der die Homepage besucht, soll die Möglichkeit haben, durch den Katalog blättern zu können.

Ein Menü soll am linken Rand der Seite zur Verfügung stehen. Dieses Menü bietet die erforderlichen Grundfunktionen an. Falls gewünscht, wird ein Untermenü am oberen Rand der Seite angezeigt.

Es soll darüber hinaus angenommen werden, daß die Multipart AG nur Produkte als Inhalte auf der WebSite darstellen möchte. Demzufolge soll auch nur der zugehörige GateType Produkt verwendet werden.

Es ist weiterhin erforderlich, daß für die Produkte neben dem DM-Preis auch ein Preis in Euro vorliegt.

Nur Einzelhändler können SiteMember, d.h. registrierte Benutzer, werden. Nur diese erhalten die Berechtigung, auch die Preise zu sehen. Nicht angemeldete Benutzer können sich via Fax mit Gewerbeanmeldung am System registrieren lassen, d.h. es soll eine diesbezügliche Registrierungsinformation angezeigt werden. Eine direkte Registrierung soll nicht möglich sein.

Einzelhändler sollen über die Web-Anwendung auch Ware bestellen können. Zusätzlich zu den standardmäßig vorhandenen GateBuilder-Funktionen (Aktionen) soll es deshalb eine Möglichkeit geben, ein Produkt in einen Warenkorb zu legen. In einem separaten Bereich namens *MyShop* soll der SiteMember seinen Warenkorb ganz oder auch nur teilweise bestellen können.

3.2 Vorgehensweise beim Aufbau einer GateBuilder-Anwendung

Es empfiehlt sich, die folgende, dem Tutorial zugrunde liegende Vorgehensweise, auch bei der Erstellung eigner GateBuilder-Anwendungen einzuhalten.

Erzeugen einer neuen GateBuilder-Anwendung

Um nicht die bereits vorhandenen GateBuilder-Anwendungen zu ändern, müssen Sie eine neue Anwendung erstellen.

Am einfachsten wird diese aus den Objekten einer vorhandenen GateBuilder-Anwendung übernommen, so daß Ihnen der Hauptaufwand bei der Konfiguration abgenommen wird. (Weitere Informationen entnehmen Sie bitte dem Abschnitt 3.3 „Erzeugen einer eigenen Anwendung“ auf S.9)

Festlegen des Layouts Ihrer neuen WebSite

Konzipieren Sie die neue WebSite nach Ihren Vorstellungen und mit Ihrem Firmen-Layout (siehe hierzu Abschnitt 3.4 „Aufbau der WebSite“ auf S.12).

Anpassen der Anwendungsumgebung

Für die neue Anwendung können die übergreifenden Standardeinstellungen, wie Pfade und Relationen vorgenommen werden. Die Benutzerumgebung kann ebenfalls angepaßt werden (weitere Informationen in Abschnitt 3.7 „Anpassung der application.cfm“ auf S.30).

Erzeugen neuer Content-Typen

Möglicherweise sind neue Content-Typen für die Darstellung der Seiteninhalte erforderlich. Diese können meist aus bereits vorhandenen GateTypes (GateBuilder spezifische Content-Typen) abgeleitet werden, sollten aber bestimmte Regeln einhalten. Sind weitere „einfache“ Content-Typen erforderlich, so sollten diese ebenfalls in dieser Phase definiert werden. (siehe Abschnitt 3.6 „Erstellen eigener Content-Typen“ auf S.22).

Umsetzen und Einbinden der Funktionalitäten (Aktionen)

Haben Sie die Content-Typen definiert, können Sie die erforderlichen Funktionalitäten (Aktionen) darauf aufbauen. Sie erstellen CustomTags für Aktionen und definieren diese in GateBuilder. (siehe 3.8 „Die Aktionen“ auf S.31).

Abbildung der Menüstruktur

Binden Sie die Aktionen in die Menüs und Toolbars ein und vergeben Sie die entsprechenden Rechte für die zugelassenen Benutzergruppen (siehe Abschnitt 3.9.1 „Die Menü-Relation“ auf S.33).

Anpassen des Designs

In einem letzten Schritt können Sie das Design Ihrer Anwendung ändern. Sie erstellen neue Ressourcen, ändern die Cascading Style Sheets und die Design-Spezifikationen der einzelnen Typen (siehe Abschnitt 3.13 „Design“ auf S.42).

3.3 Erzeugen einer eigenen Anwendung

Ziel Neue Anwendung erstellen bei gleichzeitiger Übernahme der Einstellungen aus einer bereits vorhandenen Anwendung.

3.3.1 Beschreibung

GateBuilder baut auf einer Struktur von verschiedenen Objekten auf.

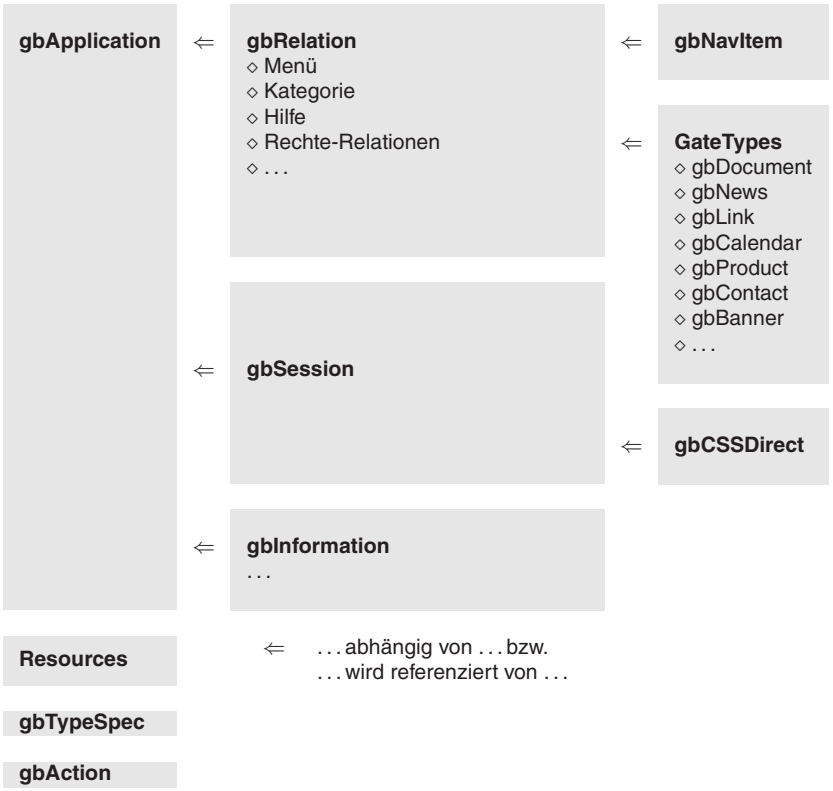

Abbildung 3.1: Struktur der Objekte

Ausgangspunkt einer Anwendung ist das Objekt `gbApplication`, welches die Eigenschaften der kompletten Anwendung speichert. Das `gbApplication`-Objekt ist sprachabhängig, d.h. für jede Sprache wird ein eigenes `gbApplication`-Objekt benötigt.

3.3.2 Beispiel

Da das angeführte Beispiel der Multipart AG nicht die Standard-GateBuilder-Umgebung benutzt, sondern ein eigenes Layout, eigene GateTypes und Ressourcen verwendet, muß eine eigene Anwendung erstellt werden (`tut_gbApplicationDE`).

3.3.3 Vorgehensweise

Um die neue Anwendung zu erstellen, starten Sie am Web-Server das Skript
`http://localhost/gatebuilder/toolbox/cloning/cloneapp.cfm`.

Dieses Skript nimmt eine vorhandene Anwendung als Vorlage für Ihre neue Anwendung und erstellt eine Kopie aller erforderlichen Objekte, Ressourcen und Dateien. Alle vom `gbApplication`-Objekt abhängigen Objekte werden ebenfalls dupliziert. Die `Content`-Objekte der alten Anwendung werden hingegen nicht übernommen.

Geben Sie für unser Beispiel folgende Parameter an. Durch Betätigen der Schaltfläche SPEICHERN starten Sie die Erzeugung einer neuen Anwendung:

Label	Eingabe	Beschreibung
Source DSN	`gatebuilder`	Name der GateBuilder-Datenbank (Quelle)
Reference GateBuilder Application	`gbApplicationDE`	Eine neue Anwendung benötigt als Vorlage eine Referenzanwendung. Diese Referenzanwendung mit ihren Relationen und Objekten wird zunächst dupliziert.
Application Name	`Tutorial`	Name der neuen GateBuilder-Anwendung
Application Path (URL)	`/tutorial/`	Virtueller Pfad unterhalb des Web-Verzeichnisses, in dem die neue Anwendung erstellt werden soll.
Target DSN	`gatebuilder`	Name der Datenquelle, in der die Anwendung erstellt werden soll.
Application Security Context	`SC1`	Verwendeter Security Context von ColdFusion
Prefix/Owner	`tut`	Um die Objekte der einzelnen Anwendungen voneinander abzugrenzen, wird dem Label der duplizierten Objekte ein Präfix vorangestellt.

Tabelle 3.1: Parameter zum Erstellen der Tutorial-Anwendung

Um die Eingaben zu speichern, klicken Sie auf die Schaltfläche SPEICHERN.

Wollen Sie die Eingaben verwerfen, klicken Sie auf die Schaltfläche ABBRECHEN.

3.3.4 Ergebnis

Es wurde das Verzeichnis `[C:\Inetpub\wwwroot\]tutorial\` in Ihrem WebRoot-Verzeichnis angelegt.

Außerdem wurden in dieses Verzeichnis, als Ausgangsbasis für die Tutorial-Anwendung, die Dateien `index.cfm`, `_left.cfm`, `_main.cfm`, `_upper.cfm` kopiert. Diese müssen für die Tutorial-Anwendung allerdings noch angepasst werden (siehe Abschnitt 3.4 „Aufbau der WebSite“ auf S.12).

In diesem Verzeichnis wurde ein neues Template `application.cfm` erstellt, das bereits gültige ObjektIDs und Namen für Ihre neue GateBuilder-Anwendung beinhaltet (siehe auch 3.7 „Anpassung der application.cfm“ auf S.30):

Im Unterverzeichnis CustomTags innerhalb des Programmverzeichnisses von GateBuilder wurde ein zusätzliches Verzeichnis `tutorial` für die neue Anwendung erstellt. In diesem sollten die CustomTags der Tutorial-Anwendung abgelegt werden.

Es wurde das `gbApplication`-Objekt `tut_gbApplicationDE` in Ihrer Datenbank angelegt und die Pfadeigenschaften dieses Objektes wurden geeignet gesetzt.

Die Relationen für Toolbars, Menüs und Kategorien wurden in Ihrer Datenbank dupliziert und das Präfix `tut` den Namen vorangestellt.

Ein `gbSession`-Objekt `tut_gbSessionDE`, welches die Standard-Benutzerumgebung definiert, wurde für die Anwendung aus `gbSessionDE` in der Datenbank erstellt (siehe auch 3.5.2 „Die Benutzerumgebung“ auf S.19).

Für die Anwendung wurden die `gbCSSDirect`-Objekte (`tut_gbCSSDirect` und `tut_gbCSSDirectALT`) in der Datenbank erstellt. Diese beinhalten die Angaben zu den Cascading Style Sheets der neuen Anwendung. (siehe auch 3.13.1 „Cascading Style Sheets“ auf S.42).

3.4 Aufbau der WebSite

Ziel Konzept und Aufbau einer GateBuilder-Anwendung kennenlernen.

GateBuilder-Funktionen in die eigene WebSite einbinden.

3.4.1 Beschreibung

Aus Gründen der einheitlichen Benutzerführung, der Benutzerfreundlichkeit und der schnelleren Entwicklung empfiehlt es sich, feste Bereiche auf ihrer WebSite für die Darstellung der einzelnen Komponenten wie Menü, Untermenü und Kategorienbaum zu definieren. So können Sie außerdem leichter auf vorhandene Komponenten zurückgreifen.

Standardmäßig wird das Template index.cfm der angegebenen Referenz-Anwendung beim Ausführen des Clone-Skripts in Ihr Web-Verzeichnis (z.B. `/Tutorial/`) kopiert. Diese Datei ist die Grundlage der gesamten Anwendung (`WebSite`). Das Layout des Templates `index.cfm` können Sie auf das gewünschte Aussehen der neuen Anwendung anpassen. In Abhängigkeit von ausgewählten Aktionen und Menüpunkten (siehe die Abschnitte 3.8 und 3.9.1) wird aus dem Template dynamisch die entsprechende HTML-Seite generiert.

3.4.2 Beispiel

Die Tutorial-WebSite (`index.cfm`) der Firma Multipart AG soll, in logische Blöcke aufgeteilt, wie folgt aussehen:

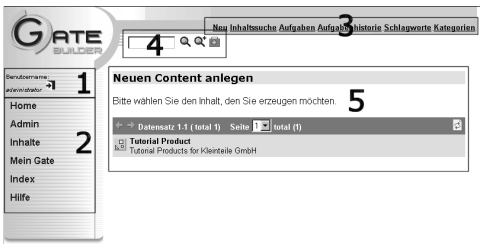

Abbildung 3.2: Layout der Tutorial-Site (Template index.cfm)

Ziffer	CustomTag [1]	Erklärung
1	`<cf_gbLogout>`	Logout
2	`<cf_gbMenu> ...`	Menü
3	`<cf_gbMenu ...`	SubMenu
4	`<cf_gbSearch>`	QuickSearch
5	`<cf_gbGate>`	Hauptfensterfunktionalität, die zentral durch Aktionen dargestellt wird., z.B. Kategorienübersicht oder Fulldisplay.

Tabelle 3.2: Erläuterungen zun Abbildung 3.2

[1] Die CustomTags definieren den Aufbau der WebSite, die mit GateBuilder aufgebaut werden soll.

3.4.3 Vorgehensweise: Aufbau des Templates index.cfm

Die logische Aufteilung der Tutorial-Anwendung soll jener der GateBuilder-Anwendung entsprechen, d.h. es existiert jeweils eine Include-Datei für den linken (`_left.cfm`) und eine für den oberen (`_upper.cfm`) Rand des Templates. Außerdem wird der Hauptbereich (`_main.cfm`) in das Template `index.cfm` eingeschlossen.

Die Datei `[c:\inetpub\wwwroot\tutorial\]index.cfm` sollte wie folgt aussehen:

```
<cfsetting enablecfoutputonly="yes">
<cf_gbformatting DataSource="#Application.Gatebuilder.DataSource#">
    <cfoutput>
        <table width="770" cellspacing="0" cellpadding="0" border="0"
        valign="top" align="left">
        <tr><td colspan="2">
    </cfoutput>
        <!-- table with logos and servicebar -->
        <cfinclude template="_upper.cfm">
        <cfoutput></td></tr>
            <tr><td valign="top" align="right">
            <table border="0" width="159" cellspacing="0" cellpadding="0" ↩
                align="left"
            valign="top" background="#application.GateBuilder[session.gatebuilder.↩
                language].AppRootURL#images/system/layout/border_left.gif">
            <tr><td valign="top">
        </cfoutput>
        <!-- Left Part with MainMenu -->
        <cfinclude template="_left.cfm">
        <cfoutput>
        </td></tr>
        <tr><td valign="bottom">
        <img src="#application.GateBuilder[session.gatebuilder.language]. ↩
            AppRootURL#images/System/layout/border_bottom.gif"></td></tr>
        </table>
        </td><td valign="top">
    </cfoutput>
        <!- main-window with gbGate ->
        <cfinclude template="_main.cfm">
    <cfoutput>
    </td></tr>
    </table><br clear="all"></cfoutput>
<!-- Writting the HTML EndTags -->
</cf_gbformatting>

<cfsetting enablecfoutputonly="no">
```

Tabelle 3.3: Das Template index.cfm

(**Anmerkung**: Auf das CustomTag `gbFormatting` im obigen Code wird im Abschnitt 3.13.1 „Cascading Style Sheets“ auf S.42 genauer eingegangen.)

3.4.4 Vorgehensweise: Der linke Rand der Seite

Der linke Rand des Templates (`_left.cfm`) soll das Menü abbilden und eine Möglichkeit zum Logout für angemeldete Benutzer bieten.

Hierzu verwenden Sie die aufgeführten CustomTags (siehe auch Tabelle 3.5):

`<cf_gblogout>`	Dieses CustomTag zeigt das Control zur Abmeldung von der Anwendung. Das Control wird nur angemeldete Benutzer.
`<cf_gbmenu>`	Dieses CustomTag erzeugt das Hauptmenü. Das Hauptmenü der Anwendung paßt sich automatisch an die Berechtigungen des einzelnen Benutzers an. Alle für die Anwendung definierten Menüpunkte sollen untereinander, also im PORTRAIT-Modus dargestellt werden.

Tabelle 3.4: CustomTags für den linken Rand der Seite

Passen Sie den linken Rand der HTML-Seite wie folgt an. Die Datei `_left.cfm` finden Sie im Web-Verzeichnis der Anwendung Tutorial `[c:\inetpub\wwwroot\]tutorial\_left.cfm`.

```
<cfsetting enablecfoutputonly=”yes”>
<cfoutput>
   <table width=”100%”>
   <tr><td><div style=”font-size:xx-small”>
</cfoutput>
<!-- Display the Logout Control -->
<cf_gbLogout>
<cfoutput>
   </div></td></tr><tr><td>
</cfoutput>
<!-- Display the MainMenu -->
<cf_gbMenu Mode=”PORTRAIT”
           RootID=”#Application.Gatebuilder[Session.Gatebuilder.Language]. ↩
              gbMenuRootID#”>
<cfoutput>
   </td></tr></table>
</cfoutput>
<cfsetting enablecfoutputonly=”no”>
```

Tabelle 3.5: Der linke Rand

3.4.5 Vorgehensweise: Der obere Rand der Seite

Die zu bearbeitende Datei heißt `_upper.cfm` und befindet sich im Verzeichnis
`[c:\inetpub\wwwroot\gatebuilder\]tutorial`.

Im oberen Bereich der HTML-Seite sollen das jeweilige Untermenü und eine Suchmaske für die Schnellsuche aufgenommen werden. Diese Funktionalitäten können mit folgenden Standard-CustomTags verwirklicht werden:

`<cf_gbmenu>`	Dieses CustomTag stellt Untermenüpunkte, die vom selben übergeordneten Menüpunkt (MenuNodeID) abhängen, nebeneinander dar, d.h. im LANDSCAPE-Mode.
`<cf_gbsearch>`	Dieses CustomTag zeigt ein Eingabefeld für die Schnellsuche und die Schaltflächen für AUSFÜHREN DER SUCHE, ERWEITERTE SUCHE und HILFE ZUR SUCHE an, so daß von hier aus alle Suchfunktionen gestartet werden können.

Tabelle 3.6: CustomTags für den oberen Rand der Seite

Der obere Rand der HTML-Seite, der durch die Datei `[c:\tutorial\wwwroot\]tutorial\_upper.cfm` definiert wird, ist wie folgt anzupassen:

```
<cfoutput>
   <table width="798" cellspacing="0" cellpadding="0" border="0" valign="top">
   <tr><td valign="top" rowspan="4" width="200" height="109"
   background="#application.GateBuilder[session.gatebuilder.language] ↩
       .AppRootURL#images/system/layout/gb_logo.gif" border=0> </td>
   <td valign="top" colspan="3">
       <table cellpadding="0" cellspacing="0" border="0">
       <tr><td align="left" valign="top" height="46" width="79"
       background="#application.GateBuilder[session.gatebuilder.language] ↩
           .AppRootURL#images/system/layout/border_upper1.gif"> </td>
       <td valign="top" align="left" height="46" width="239"
       background="#application.GateBuilder[session.gatebuilder.language] ↩
           .AppRootURL#images/system/layout/border_upper2.gif"> </td>
       <td valign="top" align="right" height="46" width="263"
       background="#application.GateBuilder[session.gatebuilder.language] ↩
           .AppRootURL#images/system/layout/border_upper3.gif">
</cfoutput>
<!-- Display the SubMenu depending on MenuNodeID -->
<cf_gbMenu
         RootID="#Application.GateBuilder[Session.GateBuilder.Language] ↩
             .gbMenuRootID#"
         NodeID="#Request.GateBuilder.UIController.MenuNodeID#"
         Type="gbSubMenu">
<cfoutput>
   </td></tr>
   <tr><td colspan="3" valign="top" height="30">
   <table width="100%">
       <tr><td width="200">
       <!-- Search Control -->
</cfoutput>
<cf_gbSearch>
<cfoutput>
   </td><td align="right"> </td>
   </tr></table>
```

```
    </td></tr></table>
    </td></tr></table>
</cfoutput>
```

Tabelle 3.7: Der obere Rand der Seite

3.4.6 Vorgehensweise: Der Hauptteil

Für den Hauptteil der HTML-Seite (`_main.cfm`) wird folgendes CustomTag verwendet:

`<cf_gbgate>`	Dieses CustomTag übernimmt die Hauptarbeit der gesamten Anwendung, d.h. es ist die zentrale Schaltstelle. Es benutzt dabei die Anweisungen und Informationen des UI-Controllers und stellt die angeforderten Inhalte im angegebenen Bereich dar.

Tabelle 3.8: CustomTags für den Hauptteil der Seite

Die Include-Datei `[c:\inetpub\wwwroot\]tutorial\_main.cfm` sollte bereits dem folgenden Layout entsprechen:

```
<cfoutput>
    <table cellspacing="0" cellpadding="0" border="0" width="10" align="left">
        <tr><td> </td></tr>
    </table>
    <table cellspacing="0" cellpadding="0" border="0" width="595" align="left">
        <tr><td>
</cfoutput>
<!-- The central work of the site is done here! -->
<cf_gbgate>
<cfoutput>
    </td></tr></table>
</cfoutput>
```

Tabelle 3.9: Der Hauptteil

3.4.7 Ergebnis

Die Dateien `index.cfm`, `_upper.cfm`, `_left.cfm` und `_main.cfm` im Web-Verzeichnis wurden überarbeitet, so daß sich das Layout dem der Beispiel-Anwendung annähert.

Wenn Sie in Ihrem Browser die Adresse `http://localhost/tutorial/index.cfm` eingeben, können Sie das Ergebnis der vorangegangenen Schritte überprüfen.

In GateBuilder sind zwei Standardsprachen definiert. In der Tutorial-Anwendung wird allerdings nur eine Sprache verwendet, was zu einer Fehlerausgabe beim Starten der Anwendung führt. Sie können diese Fehlermeldung ignorieren.

Für die Menüs der Tutorial-Anwendung wurden bisher keine Berechtigungen festgelegt. Sie können sie deshalb nur als Administrator sehen.

3.5 Konfiguration der Anwendung

3.5.1 Die Anwendungsumgebung

Ziel Eine Anwendung erfolgreich konfigurieren.

Einen Überblick über die Funktionen von `gbApplication`-Objekten gewinnen.

3.5.1.1 Beschreibung

Die Grundeinstellungen, die für die Anwendung erforderlich sind und sich auf die gesamte Anwendung beziehen, werden im `gbApplication`-Objekt hinterlegt.

Folgende Einstellungen können hier für die gesamte Anwendung vorgenommen werden:

- verschiedene Relationen (Menüs, Kategorien, Hilfe,...)
- Pfade und Laufwerk-Mappings (Upload, Web- und Programmpfad,...)
- Konfiguration der Registrierung
- Informationsobjekte
- Standard-Benutzerumgebung (`gbSession`)

Im `gbApplication`-Objekt hinterlegte Einstellungen werden beim ersten Start der jeweiligen Anwendung in den Cache geladen (siehe auch Abschnitt 3.7 „Anpassung der application.cfm“ auf S.30).

Alle Eigenschaften der Anwendung finden Sie im `Application-Scope`, welcher sprachabhängig ist:
`application.gatebuilder[session.gatebuilder.language].Eigenschaftsname=....`

Die Sprache wird benutzerabhängig im `Session-Scope` gespeichert:
`session.gatebuilder.language`.

3.5.1.2 Beispiel

Für den Zugriff auf die Inhalte der Anwendung ist eine Offline-Anmeldung per Fax (Gewerbeanmeldung) notwendig, da nur registrierte Benutzer (SiteMember) Zugriff erhalten sollen. Eine Online-Registrierung soll in der Demo-Anwendung nicht angeboten werden.[2]

Weitere Informationen zur Registrierung von Benutzern finden Sie im Administratoren-Handbuch unter den Stichworten ⇒ Registrierung oder ⇒ Bearbeiten von Anwendungen.

Unsere Anwendung wurde zwar beim Duplizieren der Referenz-Anwendung automatisch konfiguriert, jedoch sind noch folgende Grundeinstellungen zu ändern:

- Die Registrierung als SiteMember soll nicht automatisch möglich sein.
- Der Benutzer erhält eine Registrierungsinformation mit Faxnummer.

[2] Natürlich bietet GateBuilder auch die Möglichkeit einer Online-Registrierung.

3.5.1.3 Vorgehensweise

Im Anwendungsobjekt werden neben der Definition des verwendeten Menüs, des Kategorienbaums und den Toolbars (siehe auch Abschnitt 3.9 „Relationen für Menü und Kategorienbaum“ auf S.33; im Administratoren-Handbuch unter dem Stichwort ⇒ Bearbeiten von Relationen) auch die Registrierung und wichtige Anwendungspfade voreingestellt.

Mit den folgenden Parametern des `gbApplication`-Objektes lassen sich die entsprechenden Einstellungen vornehmen:

Label	Eingabe	Beschreibung
Login/Registrierung		
RegistrationShowInfo	JA	Anzeige einer Informationsseite mit Regeln und Informationen zur Nutzung des Systems.
RegistrationAskAccount	NEIN	Abfrage von Benutzernamen und Kennwort ein- oder ausschalten.
RegistrationAskUserData	NEIN	Eingabe der Benutzerdaten ein- oder ausschalten. Registrierungsdialog
Information		
RegistrationInfo	tut_gbRegistration (DE)	Informationsobjekt, das angezeigt wird, wenn RegistrationShowInfo auf JA gesetzt ist.

Tabelle 3.10: Einstellung der Registrierung

Um die Einstellungen ändern zu können, gehen Sie wie folgt vor:

- Starten Sie die Anwendung mit `http://localhost/tutorial/index.cfm`.
- Melden Sie sich als Administrator an Ihrer neuen Anwendung an.
- Klicken Sie auf das Menü **Admin**.
- Klicken Sie auf den Menüpunkt ***Konfiguration***.
- Klicken Sie den Untermenüpunkt ***Applikationen***.
- Klicken Sie auf die Schaltfläche BEARBEITEN für die Anwendung `tut_gbApplicationDE`.

Nun können Sie die Konfiguration Ihrer Anwendung vornehmen. Geben Sie die Parameter wie folgt ein:

Feld	Eingabe
`ObjectID`	`41B65781-2CC2-11D5-B36B005004F633FB`
`Label`	`tut_gbApplicationDE`
`Sprache`	`Deutsch`
Relations	
`gbMenuRootID`	`tut_gbMenuID`
`gbToolbarRootDE`	`tut_gbToolbarDE`
`gbHelpRootID`	`tut_gbHelpDE`
`gbHelpContextRootDE`	`gbHelpContextDE`
`gbCategoryRootDE`	`tut_gbCategoryDE`
`gbWorkflowExecRootID`	`gbWorkflowExec`
`gbWorkflowInitRootID`	`gbWorkflowInit`

Feld	Eingabe
Login/Registration	
LoginIsRegistration	Ja
LoginIsMandatory	Ja
RegistrationDirectAccess	Ja
RegistrationShowInfo	Ja
RegistrationAskAccount	Nein
RegistrationAskUserData	Nein
RegistrationFields	Vorname Nachname Firma Adresse PLZ
Information	
RegistrationInfo	tut_gbRegistration (DE)
ContactInfo	tut_gbFooter (DE)
eMail2FriendInfo	tut_gbFooter (DE)
Paths/Mappings	
AppRootUrl	/tutorial/
WebPath	c:\intepub\webroot\tutorial
ProgPath	c:\gatebuilder\
AppServerMapping	/GateBuilder/
UploadPath	c:\gatebuilder\uploads
Others	
EMAIL	(Tragen Sie hier eine gültige eMail-Adresse ein.)
GBSESSIONID	tut_gbSessionDE
KEYWORDCATEGORYID	gbKeywords
MENUHOME	
USERLOG	
Cache aktualisieren	(Feld aktivieren.)
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabe abbrechen
HILFE	Online-Hilfe aufrufen

Tabelle 3.11: Konfiguration der Anwendung

- Klicken Sie auf die Schaltfläche SPEICHERN, um die Änderungen zu übernehmen. Mit der Option *Cache aktualisieren* können Sie die Änderungen direkt in den Cache übernehmen und damit sofort aktivieren. Wählen Sie diese Option nicht, wird die Änderung erst beim nächsten Start des ColdFusion-Servers aktiviert.

TIP Änderungen in den Registrierungseinstellungen können Sie durch Anklicken des Links *Registrieren* in der User-Console überprüfen.

3.5.2 Die Benutzerumgebung

Ziel Bedeutung von `gbSession`-Objekten kennenlernen.

Zusammenhang zwischen den Objekten `gbApplication` und `gbSession`. erklären.

Personalisierte Benutzerumgebung erstellen.

Standard-Anwendungsumgebung anpassen.

3.5.2.1 Beschreibung

Die Anwendungsumgebung eines Benutzers kann mittels eines `gbSession`-Objektes konfiguriert werden. Dem Anwender können hier neben Sprache und Cascading Style Sheet unter anderem auch persönliche Einstellungen zugewiesen werden. Das `gbSession`-Objekt dient somit u.a. der Personalisierung.

Im `gbApplication`-Objekt kann ein `gbSession`-Objekt selektiert werden. Dieses `gbSession`-Objekt (hier: `tut_gbSessionDE`) beschreibt die Standardumgebung eines nichtangemeldeten Seitenbesuchers.

Ein SiteMember, der sich am System anmeldet, erhält nach der Registrierung und Eingabe seiner persönlichen Einstellungen, eine personalisierte Benutzerumgebung.

Das `gbSession`-Objekt, auf das in der Anwendung referenziert wird, legt neben der Sprache auch das verwendete Cascading Style Sheet fest. Die gewünschte Sprache stellt der Benutzer in seinen persönlichen Einstellungen ein. Außerdem lässt sich hier ein benutzerabhängiges Design festlegen.

Für jede Anwendung kann festgelegt werden, welche GateTypes (Eigenschaft `lUserSiteGateTypes`) verwendet werden dürfen. Diese Definition wird ebenfalls in der Standard-Benutzerumgebung (`gbSession`-Objekt) für die gesamte Anwendung vorgenommen. Während der Personalisierung werden hier die vom Benutzer bevorzugten GateTypes definiert.

3.5.2.2 Beispiel

Als Standardsprache soll Deutsch eingestellt werden.

Da es in unserem Beispiel nur möglich sein soll, mit Produkten zu arbeiten, können die GateTypes auf den Typ Produkt beschränkt werden.

Prüfen Sie zuerst, ob die Einstellungen für die Tutorial-Anwendung korrekt sind. Gehen Sie hierbei wie folgt vor:

- Melden Sie sich als Administrator an.
- Klicken Sie auf das Menü **Admin**.
- Klicken Sie auf den Menüpunkt ***Konfiguration***.
- Klicken Sie auf den Untermenüpunkt ***Applikationen***.
- Bearbeiten Sie `tut_gbApplicationDE` und überprüfen Sie die `gbSessionID` (`tut_gbSessionDE`).
- Unterbrechen Sie jetzt die Bearbeitung mit einem Klick auf die Schaltfläche ABBRECHEN.
- Klicken Sie auf den Untermenüpunkt ***Sessions***.
- Bearbeiten Sie die Session `tut_gbSessionDE`.
- Überprüfen Sie folgende Werte:

Language	Deutsch (⇒ DE)
CSSID	`tut_gbCSSDirect`
CSSID_Alt	`tut_gbCSSDirectAlt`

- Überprüfen Sie in der Auswahlliste `lUserSiteGateTypes`, daß ausschließlich der Punkt *Produkte* ausgewählt ist.
- Klicken Sie die Schaltfläche SPEICHERN, um die Änderungen zu übernehmen.
- Die Änderungen müssen jetzt noch in den Cache übernommen werden. Öffnen Sie die Bearbeitungsmaske der Anwendung `tut_gbApplicationDE` erneut und aktivieren Sie die Checkbox *Cache aktualisieren*. Klicken Sie danach auf die Schaltfläche SPEICHERN (siehe auch 3.3 „Erzeugen einer eigenen Anwendung“ auf S.9).

Feld	Eingabe
ObjectID	41B5683F-2CC2-11D5-B36B005004F633FB
Label	tut_gbSessionDE
Language	Deutsch
lUserSiteGateType	Produkte
lUserSiteCategories	
UserSiteTimeSpan	-unbegrenzt-
lUserNotifyGateType	Produkte
lUserNotifyCategories	
UserNotifyFrequencies	Täglich
UserNotifyMethod	per eMail
LastNotifyDateTime	-
bNewsletterSubscribed	(Feld aktivieren.)
lNewsletterCategories	
lfavoriteObjectIDs	
lcommentedTypeIDs	Produkte
CSSID	gbCSSDirect
CSSID_ALT	gbCSSDirectALT
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabe abbrechen
HILFE	Online-Hilfe aufrufen

Tabelle 3.12: Konfiguration der Benutzerumgebung mit gbSession-Objekten

3.5.2.3 Ergebnis

Diejenigen Standardeinstellungen der Anwendung, welche die Benutzerumgebung betreffen, wurden in der Session vorgenommen. Dies sind:

- Sprache
- GateTypes
- Cascading Style Sheets

Darüber hinaus ist es jetzt nur noch möglich, Produkte als Inhalt zu stellen. Wenn Sie sich als Administrator an Ihrer Tutorial-Anwendung anmelden und Inhalte anklicken, erhalten Sie somit auch nur Produkte als mögliche GateTypes.

3.6 Erstellen eigener Content-Typen

3.6.1 Eigene GateTypes

Ziel Neue GateTypes ertellen.

Neue GateTypes in die Anwendung einbinden.

3.6.1.1 Beschreibung

GateTypes, also Content-Typen für die Darstellung von Inhalten auf der WebSite, müssen in GateBuilder folgende Bedingungen erfüllen, damit sie einheitlich und korrekt von den GateBuilder-CustomTags verarbeitet werden können.

Name	PropertyType	Beschreibung
Benötigte Eigenschaften:		
PublishInfo	PublishInfo	Diese Struktur enthält alle Informationen, die für den internen Ablauf und die Veröffentlichung erforderlich sind.
Es empfiehlt sich, folgende oft benötigte Eigenschaften ebenfalls zu verwenden:		
Title	Title	Titel oder Name des Objektes
SubTitle	SubTitle	Untertitel, der meist in Listen verwendet wird
Teaser	Teaser	Formatierte Kurzbeschreibung
Body	Body	Ausführlicher Text, Hauptteil

Tabelle 3.13: Eigenschaften eines GateTypes

Name	Beschreibung
Activate	Setzt das Objekt aktiv, um im Web veröffentlicht zu werden.
Archive	Archiviert das Objekt.
Create	Anlegen einer leeren Objektstruktur.
Delete	Führt die Funktionalitäten aus, welche zusätzlich vor dem Löschen des Content-Objekts ausgeführt werden sollen.
Display	Vollanzeige (Fulldisplay) des Inhalts mit Toolbar und evtl. erweiterter Information für Editoren, Contributoren und Administratoren.
Edit	Wiedergabe des Inhalts mit der Möglichkeit, diesen zu bearbeiten.
ListDisplay	Handler für die Ausgabe von Listen (CustomTag `gbListControl`).
PrintDisplay	Druckfähige Anzeige, die zum Teil in andere Display-Handler eingebunden wird.
Save	Handler wird beim Speichern aufgerufen, um weitere von diesem Content-Typ abhängige Arbeiten, wie Upload oder das Speichern von Substrukturen, durchzuführen.
TaskDisplay	Anzeige des Objektes in der Aufgabenliste.
TopDisplay	Anzeige des Objektes als Top-Content auf der HomePage.
Validate	Prüft, ob alle Eigenschaften des Content-Objekts die für die Veröffentlichung erforderlichen Werte besitzen.

Tabelle 3.14: Benötigte Handler für GateTypes

3.6.1.2 Beispiel

In der Anwendung der Multipart AG ist es erforderlich, daß der Produkttyp (`gbProduct`) um eine Preisangabe in EURO (PriceEuro) erweitert wird.

Die Display-Handler sollen dahingehend angepaßt werden, daß die Preise nur für SiteMember angezeigt werden.

3.6.1.3 Vorgehensweise

Im Anschluß an diese Aufzählung wird auf die Schritte näher eingegangen.

- Duplizieren Sie den vorhandenen GateType `gbProduct`.
- Passen Sie den neuen GateType Ihren Anforderungen an.
- Ergänzen Sie die Handler des neuen `GateTypes`.
- Passen Sie abschließend die erlaubten GateTypes für die Anwendung an.

3.6.2 Duplizieren eines vorhandenen GateTypes

Ziel Einen neuen GateType aus einem bereits Bestehenden erzeugen.

3.6.2.1 Beschreibung

GateBuilder liefert Ihnen bereits vordefinierte GateTypes zur Darstellung von Bannern (Werbeflächen), Dokumenten, Kontaktadressen, Links, News, Produkten, Newslettern und Terminen.

Anstatt einen völlig neuen GateType zu erstellen, können sie auch einen bereits vorhandenen GateType, dessen Struktur ähnlich ist, einfach duplizieren.

Das Skript `http://localhost/gatebuilder/toolbox/cloning/clonegatetype.cfm` hilft Ihnen dabei. Sie sparen dadurch unnötige Schreibarbeit und vermeiden Fehlerquellen.

Folgende Aufgaben erledigt das Skript für Sie:

- Kopieren der Eigenschaften des Referenz-GateTypes in den neuen GateType.
- Kopieren der Handler des Referenz-GateTypes.
- Generieren der Ressourcen für die Bezeichnung des neuen GateTypes.
- Generieren der neuen `gbTypeSpec`-Objekte für den neuen GateType (näheres dazu im Abschnitt 3.13.4 „Typ-Spezifikation“ auf S.48).
- Eintragung des GateTypes in die Liste der vorhandenen GateTypes (PropertyDefinition `gbGateType`).

3.6.2.2 Beispiel

Die Firma Multipart AG benötigt für ihre Anwendung einen GateType, der es ermöglicht, Produkte mit zwei Preisen (DM und EURO) darzustellen. Außerdem sollen nur die Benutzer Preise sehen, die zur Gruppe SiteMembers (Einzelhändler) gehören.

Zum Kopieren des GateTypes Products starten Sie das Skript
`[http://localhost]/gatebuilder/toolbox/cloning/clonegatetype.cfm`.

Geben Sie die folgenden Parameter an:

Parameter	Eingabewert	Beschreibung
Source DataSource	`gatebuilder`	Name der Datenquelle, in welcher sich der Referenz-GateType befindet.
Destination DataSource	`gatebuilder`	Name der Datenquelle für den neuen Content-Typ
Reference GateType	`Produkte`	GateBuilder-Content-Typ (GateType), der dupliziert werden soll.
Prefix for new GateType	`tut`	Dieses Präfix wird der Bezeichnung Ihres neuen GateTypes vorangestellt.
Relative path for handler root	`[/gatebuilder/]handlers/`	Relativer Suchpfad für Allaire Spectra
Absolute path for handler root	`[c:\gatebuilder\]handlers\`	Absoluter Suchpfad für Allaire Spectra

Tabelle 3.15: Parameter zum Duplizieren des GateTypes Products

- Beenden Sie die Eingabe der Daten durch einen Klick auf die Schaltfläche SPEICHERN.

3.6.3 Anpassen des GateTypes

Ziel Bestehende GateTypes anpassen.

Neue GateTypes in Allaire Spectra erzeugen.

3.6.3.1 Beschreibung

GateTypes werden normalerweise direkt in Allaire Spectra definiert. Wenn Sie das Skript zum Duplizieren eines GateTypes verwenden, müssen Sie nur noch die notwendige Erweiterung der Eigenschaften des neuen GateTypes vornehmen.

3.6.3.2 Beispiel

In der Tutorial-Anwendung soll neben dem Preis in DM auch ein Preis in EURO (PriceEURO) angegeben werden.

3.6.3.3 Vorgehensweise

Gehen Sie hierzu in Allaire Spectra wie folgt vor:

- Starten Sie den Webtop von Allaire Spectra.
- Melden Sie sich als Administrator an.
- Wählen Sie in der Auswahlbox *Current Application* den Eintrag Tutorial aus.
- Klicken Sie auf *Types and Objects*.
- Wählen Sie *Types* aus.
- Suchen Sie den Content-Typ `tutgbProduct`, welcher durch das Clone-Script angelegt wurde.
- Klicken Sie nun auf das Symbol Edit.
- Klicken Sie im Bereich Properties auf die Schaltfläche NEW.
- Ergänzen Sie die folgende Objekt-Eigenschaft für den neuen GateType `tut_gbProduct`.
  - Geben Sie unter *New Property Name*: PriceEURO ein und wählen Sie in der Auswahlbox *Property Definition* den Eintrag „text“ aus.
  - Klicken Sie auf die Schaltfläche NEXT.
- Wählen Sie die Option *Required* an , so daß die Eigenschaft immer vorhanden ist.
- Klicken Sie auf die Schaltfläche CREATE, um die Eigenschaftserfassung zu beenden.

3.6.3.4 Ergebnis

Der neue GateType wurde nun den Anforderungen der Multipart AG entsprechend angepaßt.

3.6.4 Erweitern der Handler

Ziel Änderungen in den Handlern von GateTypes vornehmen.

3.6.4.1 Beschreibung

Die neuen Eigenschaften des geänderten GateTypes müssen zuerst in die einzelnen Handler der Methoden, welche vom Referenz-GateType kopiert wurden, aufgenommen werden, damit die Methoden des neuen GateTypes ebenfalls mit diesen arbeiten können.

Die Handler des neuen GateTypes finden Sie im Pfad `[c:\gatebuilder\]handlers\tutgbProduct\`.

3.6.4.2 Vorgehensweise: create.cfm

Alle Eigenschaften eines Objektes sollten beim Erzeugen des Objektes vorhanden sein und initialisiert werden. Fügen Sie hierzu die neue Eigenschaft der Liste der zu initialisierenden Eigenschaften hinzu.

Beispiel

Die Änderung in der Datei `create.cfm` des GateTypes `tutProducts` (`tut_gbProduct`) könnte wie folgt aussehen:

Die Schleife muß um die Initialisierung der Eigenschaft PriceEURO - wie in der folgenden Abbildung dargestellt - erweitert werden.

```
...

<!-- Initialize the Product Object itself -->
<cfloop list="Title,SubTitle,Teaser,Price,PriceEURO,Body" index="attr">
<cfset stObject["#attr#"] = " ">
</cfloop>

...
```

Tabelle 3.16: Erweiterung des Handlers create.cfm

3.6.4.3 Vorgehensweise: editstep.cfm

Zum Bearbeiten eines GateType-Objekts wird der vordefinierte Prozess (PLP) `gbPublish` gestartet, der die einzelnen erforderlichen Schritte der Bearbeitung definiert. Wollen Sie eigene Schritte in den Ablauf aufnehmen, können Sie diese im Edit-Handler (`edit.cfm`) des GateTypes definieren. Für Änderungen am Veröffentlichungsprozess eines GateTypes sind nur Änderungen an den Methoden des jeweiligen GateTypes erforderlich. Der Handler EditStep ist für die GateBuilder-GateTypes der erste Schritt des Prozesses und ermöglicht es, die Hauptinformation des Inhaltes zu bearbeiten. (Weitere Informationen finden Sie in der Spectra-Dokumentation).

Beispiel

Fügen Sie das Eingabefeld für den EURO-Preis hinzu. Desweiteren müssen Sie den Wert in den ControlEvent-Handlern, wie in Abschnitt 3.6.4.2 beschrieben, initialisieren und speichern.

Fügen Sie den mit **!** gekennzeichneten Code in den Handler `editstep.cfm` ein, welchen Sie im Verzeichnis `[c:\gatebuilder\]handlers\tutgbproduct\` finden.

```
  <TR><!-- Price in DM -->
! <TD class="TDLabel"><cf_gbtext Resource_ID="407"> (DM)</TD>
  <TD><INPUT
        type="Text"
        name="#cn#.Price"
        size=10
        value="#cd.Price#">
  </TD>
  <TD></TD>
  </TR>

! <TR><!-- Price in EURO -->
! <TD class="TDLabel"><cf_gbtext Resource_ID="407"> (EURO)</TD>
! <TD><INPUT
!       type="Text"
!       name="#cn#.PriceEURO"
!       size=10
!       value="#cd.PriceEURO#">
! </TD>
! <TD></TD>
! </TR>
```

Tabelle 3.17: Änderungen in der EditStep-Methode

3.6.4.4 Vorgehensweise: printdisplay.cfm

Um die neuen Eigenschaften des Objektes dem Benutzer zugänglich zu machen, müssen diese in die Methode `PrintDisplay` des GateTypes aufgenommen werden. Die Erweiterung von PrintDisplay, hat zur Folge, daß die Eigenschaft in allen Anzeige-Methoden sichtbar ist, da diese die PrintDisplay-Methode zur Darstellung des Inhaltes aufrufen.

Beispiel

Nur angemeldete Benutzer sollen die Preise angezeigt bekommen. Anhand der Security-Einträge im Session-Scope können Sie prüfen, ob der jeweilige Benutzer angemeldet ist.

In der Struktur `Session.GateBuilder.Security` sind folgende Einträge für angemeldete Benutzer bekannt:

Schlüssel	Beschreibung
Username	Login-Name des Benutzers.
Fullname	Der vollständige Name des Benutzers.
Password	Das Kennwort des Benutzers.
lGroups	Liste aller Gruppen, welcher der Benutzer angehört.

Tabelle 3.18: Session.GateBuilder.Security

Im folgenden Beispiel wird auf eine Angabe registrierter Benutzer verzichtet, um einer anwenderspezifischen Definition von eigenen Gruppen nicht vorzugreifen. Es wird vielmehr davon ausgegangen, daß ein registrierter Nutzer (SiteMember) immer den Usernamen im `SessionScope` mitführt.

Wie im PrintDisplay-Handler die Benutzerdaten abgefragt werden, sehen Sie im Code-Beispiel der Tabelle 3.19:

```
<!-- CHECK, IF THE USER IS ALLOWED TO SEE THE PRICES -->
<cfif isDefined("Session.GateBuilder.Security.Username") and↩
   len(Session.GateBuilder.Security.Username)>
   <cfif len(gbProduct.Price)>
      <tr><td colspan=2> </td></tr>
      <tr><td class="TDLabel"><cf_gbtext Resource_ID=407> (DM)   </td>
      <td>#gbProduct.Price#</td></tr>
      <tr><td class="TDLabel"><cf_gbtext Resource_ID=407> (EURO) </td>
      <td>#gbProduct.PriceEURO#</td></tr>
   </cfif>
</cfif>
```

Tabelle 3.19: Erweiterung des Handlers printdisplay.cfm

3.6.4.5 Übung

Starten Sie den ColdFusion-Server neu, damit die Änderungen aktiv werden.

Der neue GateType `tutProdukte` (`tutgbProduct`) soll nun in der Tutorial-Anwendung als einziger GateType verwendet werden. Nehmen Sie dazu die notwendigen Änderungen im `gbSession`-Objekt vor und lesen Sie die Anwendungsparameter in den Cache ein (siehe hierzu auch 3.5.2 „Die Benutzerumgebung“ auf S.19).

3.6.4.6 Ergebnis

Der GateType verfügt jetzt über alle notwendigen Handler und ist damit vollständig in die Tutorial-Anwendung integriert.

3.6.5 Content-Typen unter Spectra anlegen

Ziel Weitere Content-Typen in Allaire Spectra anlegen.

3.6.5.1 Beschreibung

Für die Erstellung neuer Funktionalitäten können weitere Content-Typen erforderlich sein. Neue Content-Typen können nur mittels Allaire Spectra definiert werden.

3.6.5.2 Beispiel

In unserem Beispiel ist ein Warenkorb für die Erfassung von Aufträgen erforderlich.

Dieser Warenkorb (`tutShoppingCart`) enthält als Verweis auf den Benutzer den Benutzernamen und eine Struktur mit der ObjectID und die Anzahl der zu bestellenden Produkte (`tutgbProduct`-Objekte).

3.6.5.3 Vorgehensweise

Legen Sie den Content-Typ `tutShoppingCart` wie folgt in Allaire Spectra an:

- Starten Sie den Webtop von Allaire Spectra.
- Melden Sie sich als Administrator an.
- Klicken Sie auf *Types and Objects*.
- Wählen Sie *Types*.
- Klicken Sie auf die Schaltfläche CREATE, um den neuen Content-Typ zu erstellen.
- Geben Sie folgende Parameter ein:

Eigenschaft	Parameter
Label	`tutShoppingCart`
Description	the Shopping Cart for the tutorial application
Handler Root	`/gatebuilder/handlers/tutshoppingcart/`

Tabelle 3.20: Parameter für Content-Typ tutShoppingCart

- Klicken Sie auf die Schaltfläche OK.
- Klicken Sie auf die Schaltfläche NEW im Bereich *Properties* an.
- Legen Sie folgende Eigenschaften für das Objekt an:

Eigenschaft	Parameter	Beschreibung
Username	Text || Required	Name des Benutzers, wie er auch im UserDirectory oder UserProfile verwendet wird.
stItems	Structure* || Required	Struktur mit der Anzahl der bestellten Produkte

(* `Structure` gibt es als Parameter nur, wenn im Benutzerprofil „System Entities“ eingeschaltet ist.)

Tabelle 3.21: Eigenschaft des Content-Typs tutShoppingCart

- Klicken Sie die Schaltfläche OK, um die Erfassung der Eigenschaften zu beenden.
- Klicken Sie die Schaltfläche OK im Fenster Edit Type, um die Änderungen zu übernehmen.

3.6.5.4 Ergebnis

Dieses neue Administrationsobjekt kann nun von den entsprechenden Aktionen und Funktionen verwendet werden.

3.7 Anpassung der application.cfm

Ziel Umgang mit dem Template `application.cfm` erlernen.

Umgang mit der Initialisierung der Anwendung erlernen.

3.7.1 Beschreibung

Vor jedem Aufruf eines Templates, z.B. `index.cfm` oder `gbgateinvoker.cfm`, wird die Datei `application.cfm` ausgeführt.

Die Funktionalität des ColdFusion-Servers wird genutzt, um durch das CustomTag `gbApplicationInitialize` die Anwendung mit den erforderlichen Parametern zu initialisieren. Hierzu wird das Spectra Anwendungsobjekt Tutorial herangezogen.

Um die Einstellungen der GateBuilder-Anwendung zu definieren, wird die ObjektID des neu erstellten `gbApplication`-Objekts (`tut_gbApplicationDE`) als Parameter `lgbApplicationIDs` mitgegeben.

Für mehrsprachige Anwendungen geben Sie hier einfach die ObjectIDs der einzelnen `gbApplication`-Objekte an. Ein solches Objekt beschreibt dabei immer eine einsprachige Teilanwendung.

3.7.2 Beispiel

Erweiterungen der Anwendungsumgebung, wie z.B. allgemeine Einstellungen oder IDs von in der Anwendung verwendeten Content-Typen, können direkt in der Datei `application.cfm` vorgenommen werden.

Für das Beispiel werden definierte Variablen für die TypeIDs der neuen Content-Typen `tutgbProduct` und `tutShoppingCart` benötigt.

3.7.3 Vorgehensweise

Die TypeIDs finden Sie wie folgt in Allaire Spectra:

- Starten Sie den Webtop von Allaire Spectra.
- Melden Sie sich als Administrator an.
- Wählen Sie die Kategorie *Types and Objects*.
- Wählen Sie den Eintrag *Types*.
- Suchen Sie die Content-Typen `tutgbProduct` und `tutShoppingCart` und notieren Sie die angezeigten TypeIDs.
- Öffnen Sie die Datei `application.cfm` im Verzeichnis `[c:\inetpub\wwwroot\]tutorial\`.
- Ändern Sie die Datei `application.cfm` in Ihrem Web-Verzeichnis wie folgt ab:

```
<cfparam name="bRefreshCache" default ="FALSE">
<cf_gbapplicationinitialize
       Name = "Tutorial"
       lgbApplicationIDs ="ObjectID von tut_gbApplicationDE"
       Owner ="tut"
       Mode ="Browse" bRefreshCache ="#bRefreshCache#">
<cfset Application.Gatebuilder.tutProductTypeID="INSERT tutgbProduct-TYPEID ↩
          HERE">
<cfset Application.Gatebuilder.tutShoppingCartTypeID="INSERT tutShoppingCart- ↩
          TYPEID HERE">
```

Tabelle 3.22: Code-Beispiel: Ergänzungen zu Datei application.cfm

3.8 Die Aktionen

Ziel Eine Aktion erstellen.

Eine Aktion in der Anwendung anlegen.

3.8.1 Beschreibung

Jede Funktion, die eine GateBuilder-Anwendung ausführt, muß durch ein Ereignis ausgelöst werden. In der Regel geschieht dies dadurch, daß ein Benutzer auf der Seite eine Aktion ausführt, beispielsweise durch Anklicken eines Links oder durch Senden eines Formulars. Diese Funktionalität von Links wird bei der GateBuilder-Anwendung auch für Menüs, Toolbars oder Kategorien eingesetzt.

Jede Aktion wird als `gbAction`-Objekt in der Datenbank definiert. Über die ObjectID, bzw. über den Namen des Objekts kann eindeutig eine bestimmte Aktion, z.B. über URL-Parameter, aufgerufen werden.

`[http://localhost]/tutorial/index.cfm?gbAction=NAME_DER_AKTION` oder
`[http://localhost]/tutorial/index.cfm?gbAction=OBJECTID_DER_AKTION`

Jede Auswahl eines Menüpunktes oder einer Toolbar-Schaltfläche (`gbNavItems`) führt zum Ausführen einer Aktion.

Das Aktivieren eines Links bewirkt den Aufruf der Aktion. Durch jede Benutzeraktion wird ein bestimmtes Custom-Tag durch die Execute-Methode der Aktion (`gbAction`-Objekt) im `gbGate`-Bereich der Seite ausgeführt.

3.8.2 Beispiel

Im Moment werden in der Tutorial-Anwendung noch Top-Inhalte und die UserConsole (Login-Felder, Sprachauswahl und Registrierungslink) auf der Homepage angezeigt. Die Firma Multipart AG fordert allerdings die User-Console sowie eine Kategorieübersicht anstelle der Top-Inhalte.

3.8.3 Vorgehensweise

Um eine Änderung der Anzeige vorzunehmen, wird zuerst ein CustomTag benötigt, das die erforderliche Funktionalität anbietet.

Erstellen Sie das CustomTag `tutCategoryOverviewAndUserConsole`, welches a) die Kategorienübersicht (`cf_gbCategoryOverview`) zeigt und b) die UserConsole (`cf_gbuserconsole`) am rechten Rand aufbaut.

Hierzu können bereits vorhandene allgemeine CustomTags von GateBuilder verwenden werden.

Erstellen Sie dieses CustomTag mit dem Dateinamen `tutcategoryoverviewanduserconsole.cfm` (Pfad `[c:\gatebuilder\]customtags\tutorial\`.

```
<cfsetting enablecfoutputonly="yes">
<cfoutput>
<table width="100%" border="0"><tr><td valign="top" bgcolor="##DFDFF3">
</cfoutput>
<!-- Display the Category Overview ->
<cf_gbcategoryoverview>
<cfoutput></td><td align="right" valign="top"></cfoutput>
<!- Display the UserConsole for Login and Registration ->
<cf_gbuserconsole>
<cfoutput></td></tr></table></cfoutput>
<cfsetting enablecfoutputonly="no">
```

Tabelle 3.23: Code-Beispiel: CustomTag tutCategoryOverviewAndUserconsole

Erstellen Sie eine Aktion für dieses CustomTag. Gehen Sie wie folgt vor:

- Melden Sie sich als Administrator in der Tutorial-Anwendung an.
- Klicken Sie auf das Menü **Admin**.
- Wählen Sie den Menüpunkt ***Konfiguration***.
- Die Liste mit den Aktionen ist bereits ausgewählt. Klicken Sie auf die Schaltfläche NEU, um eine neue Aktion anzulegen.
- Legen Sie folgende Aktion an:

Feld	Eingabe
ObjectID	247A16E0-5D46-11D4-B288005004F633FB
Label	`tutShoppingCartItemAdd`
CustomTag	`tutShoppingCartItemAdd`
Parameter(s)	
`BackgroundCall`	(Feld aktivieren)
Beschreibung	(Fügen Sie hier eine Beschreibung der Aktion ein.)
Zugänglich für	SiteMembers (Feld aktivieren)
Cache aktualisieren	(Feld aktivieren)
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabe abbrechen
HILFE	Online-Hilfe aufrufen

Tabelle 3.24: Erstellen der Aktion tutCategoryOverviewAndUserConsole

- Achten Sie vor dem Speichern darauf, daß *Cache aktualisieren* ausgewählt ist, so daß die Änderungen

sofort aktiviert werden.

- Sichern Sie diese Aktion durch Betätigen der Schaltfläche SPEICHERN.

3.8.4 Ergebnis

Die Aktion `tutgbCategoryOverviewAndUserConsole` ist jetzt angelegt und aktiviert, allerdings fehlt noch eine Verbindung zwischen dem Menüpunkt **Home** und der neuen Aktion. Die Einbindung ins Menü erfolgt im nächsten Schritt.

3.9 Relationen für Menü und Kategorienbaum

3.9.1 Die Menü-Relation

Ziel Menüs erfolgreich verwalten.

Aktionen in Menüs einbinden.

3.9.1.1 Beschreibung

Die Menüstruktur wird über sogenannte GateBuilder-Relationen (`gbRelations`) realisiert.

Mit `gbRelation`-Objekten wird die Navigation innerhalb einer GateBuilder-Anwendung abgebildet (Mehr hierzu finden Sie unter dem Stichwort ⇒ Relationen im Administratoren-Handbuch). Sowohl Menüs, Kategorienbäume, als auch Toolbars werden mit Hilfe von `gbRelation`-Objekten abgebildet. Relationen werden mit Hilfe eines Root-Objektes (`gbRelation`) sowie den dazugehörigen Navigationsobjekten (`gbNavItem`) definiert. Alle Relationen sind als Baumstruktur definiert, wobei die Relationen aber auch für 1:n- oder m:n-Relationen (Beziehungen zwischen Objekten) verwendet werden können.

Die Zugriffsrechte auf einzelne `gbNavItem`-Objekte werden benutzergruppenabhängig vergeben, so daß deren Funktion nur für bestimmte Benutzergruppen entsprechend ihren Rechten sichtbar und damit auch zugänglich ist.

3.9.1.2 Beispiel

Die Menüstruktur muß für die Firma Multipart AG wie folgt angepaßt werden:

Menü/Untermenü	Änderung	Hinweise
Home	Zugriff: `_allusers` Aktion: `tutcategoryoverviewanduserconsole`	
Admin	Zugriff: Administrators	Untermenüs wie bisher, prüfen Sie die Berechtigungen
Inhalte	Zugriff: Contributor/Editor	Untermenüs wie bisher, prüfen Sie die Berechtigungen
MeinShop	Zugriff: SiteMembers	Menüpunkt **MeinGate** umbenennen
MeinShop	Zugriff: SiteMembers	Untermenüpunkt ***MeinGate*** umbenennen
Kennwort	Zugriff: SiteMembers	
Einstellungen	Zugriff: SiteMembers	
Index	Zugriff: _allusers	
Produktgruppen	Zugriff: _allusers	Menüpunkt ***Kategorien*** umbenennen
Titel	Zugriff: _allusers	
Schlagworte	Zugriff: _allusers	
Hersteller	Zugriff: _allusers	

Tabelle 3.25: Menü-Definition

3.9.1.3 Vorgehensweise

- Melden Sie sich als Administrator (GateBuilder) an.
- Klicken Sie auf **Admin**.
- Wählen Sie den Menüpunkt ***Konfiguration***.
- Klicken Sie auf den Untermenüpunkt ***Relationen***.
- Um das Menü der Tutorial-Anwendung zu bearbeiten, klicken Sie auf die Schaltfläche RELATION BEARBEITEN der Relation `tut_gbMenuDE` an.
- Passen Sie das Menü entsprechend der Definition an. Bei der Übernahme der Menü-Relation aus der Referenz-Anwendung werden keine Berechtigungen übernommen, d.h. Sie müssen ggf. sämtliche Rechte für die Menüpunkte nochmals entsprechend der Angaben überarbeiten.
  Eine genaue Beschreibung der Menüpflege finden Sie im Benutzerteil des Administratoren-Handbuches.

Klicken Sie auf die Schaltfläche ÜBERNEHMEN, um Änderungen zu übernehmen. Änderungen werden aber erst nach einem Klick auf die Schaltfläche SPEICHERN an den Server geschickt und dort gespeichert!

Möchten Sie, daß eine Aktion nicht direkt im `gbGate`-Bereich (siehe hierzu auch 3.4 „Aufbau der WebSite“ auf S.12) Ihrer aktuellen Seite ausgeführt wird, sondern in einem neuen Browser-Fenster, so können Sie durch Angabe der Eigenschaft *Fenstername/Aussehen* des Menüpunktes (`gbNavItem`) bestimmen, in welchem Frame oder Fenster Ihre WebSite angezeigt werden soll.

Soll ein neues Fenster geöffnet werden, so startet GateBuilder immer das Template
`http://localhost/gatebuilder/gbgateinvoker.cfm`,
welches wiederum das CustomTag `gbGate` aufruft (vgl. Abschnitt 3.4 „Aufbau der WebSite“ auf S.12).

3.9.2 Anlegen von Kategorien

Ziel Bedeutung von Kategorien kennenlernen.

Kategorien (Relationen) erfolgreich pflegen.

Anwendungsmöglichkeiten von Relationen kennenlernen.

3.9.2.1 Beschreibung

Ein in GateBuilder erstellter Kategorienbaum besitzt die übliche hierarchische Baumstruktur, bei der jede Kategorie wieder Unterkategorien besitzen kann. Die Seiteninhalte sind die *Blätter* dieses Baumes, wobei die Inhalte aber einer oder auch mehreren Kategorien zugeordnet werden können. Zur technischen Realisierung der Kategorien werden, wie bei den Menüs, Relationen verwendet.

Da die Zugriffsrechte auf einen Inhalt (und damit auch seine Sichtbarkeit) über die Kategorisierung gesteuert werden, müssen Sie ihm vor der Veröffentlichung im Web Kategorien zuordnen.

Für Kategorien müssen keine Aktionen hinterlegt werden, da im Gegensatz zu den Menüpunkten eine Standardfunktion aufgerufen wird.

3.9.2.2 Beispiel

Die Produktgruppen der Firma Multipart AG, die als Kategorien dargestellt werden können, sind wie folgt gegliedert:

Kategorie Unterkategorie	Beschreibung	Sortierung	Zugänglich für
Haushalt	Alles für den Haushalt: Küche, Wohnen, Bad	010	_ALLUSERS
Küche	Küchenzubehör	010	_ALLUSERS
Bad	Vom Entchen bis zur Duschhaube	020	_ALLUSERS
Wohnen	Wohnen wie Gott in Frankreich	030	_ALLUSERS
Garten	Alles für den Garten	020	_ALLUSERS
Rasen	Englischer Rasen, Wildkräuterwiesen, ...	010	_ALLUSERS
Party	It's party time.	020	_ALLUSERS
Sonderangebote	Angebote, Restposten, Broschüren, ...	030	sitemembers
Werbeartikel	Alles für Ihren Umsatz	010	sitemembers
Broschüren	Prospekte, Flyer, ...	010	sitemembers
Geschenkartikel	Kleine Geschenke erhalten die Freundschaft	020	sitemembers
Restposten	Allerlei was noch übrig ist	020	sitemembers

Tabelle 3.26: Darstellung der Produktgruppen als Kategorienbaum

3.9.2.3 Vorgehensweise

Die Produktgruppen lassen sich wie eine Kategorisierung der Produkte verstehen. Es kann also die Standardfunktion (CustomTag `gbCategoryOverview`) zur Darstellung von Kategorien verwendet werden.

- Melden Sie sich als Administrator an.
- Wählen Sie das Menü **Admin**.
- Klicken Sie auf den Menüpunkt ***Konfiguration***.
- Klicken Sie auf den Untermenüpunkt ***Relationen***.
- Klicken Sie auf die Schaltfläche ITEM BEARBEITEN der Relation `tut_gbCategoryDE`.
- Ändern Sie den Namen der Relation in *Tutorial-Produktgruppen* ab.
- Klicken Sie auf die Schaltfläche SPEICHERN.
- Um die Produkt-Kategorien der Tutorial-Anwendung zu bearbeiten, klicken Sie auf die Schaltfläche RELATION BEARBEITEN der Relation `tut_gbCategoryDE`.
- Passen Sie die Kategorien entsprechend der obigen Definition an.
  Eine weitergehende Beschreibung der Relationspflege finden Sie im Administratoren-Handbuch. Außer dem Namen und den Rechten der einzelnen Kategorie sind keine weiteren Angaben erforderlich.
- Wollen Sie eine Unterkategorie anlegen, wählen Sie die jeweilige übergeordnete Kategorie und klicken dann auf die Schaltfläche NEU.
- Klicken Sie nach der Änderung einer Kategorie auf die Schaltfläche ÜBERNEHMEN, damit die Änderungen übernommen werden.
- Klicken Sie anschließend auf die Schaltfläche SPEICHERN, um alles zu speichern.

3.9.2.4 Ergebnis

Das CustomTag `tutCategoryOverviewAndUserConsole`, welches das CustomTag `gbCategoryOverview` aufruft, wurde bereits als Aktion für den Home-Menüpunkt hinterlegt. Wenn Sie also den Menüpunkt **Home** auswählen, erhalten Sie die Übersicht der Produktgruppen.

Überprüfen Sie die Kategorien-/Produktgruppenübersicht für einen nicht angemeldeten Benutzer und für den Benutzer SiteMember. Welche Unterschiede fallen Ihnen dabei auf?

3.10 Veröffentlichen von Inhalten

Ziel Inhalte auf der WebSite veröffentlichen.

3.10.1 Beschreibung

Sie können jetzt Tutorial-Produkte (`tutgbProduct`) anlegen und so Ihre ersten Inhalte für die neue Anwendung erstellen. Eine ausführliche Beschreibung der Arbeitsabläufe beim Bearbeiten von Inhalten finden Sie im Benutzerteil des Administratoren-Handbuchs.

3.10.2 Beispiel

Im Beispiel soll ein Produkt vom Editor im Web veröffentlicht werden.

3.10.3 Vorgehensweise

Um ein neues Produkt (`tutgbProduct`) zu erzeugen, gehen Sie wie folgt vor:

- Melden Sie sich als Editor an.
- Klicken Sie auf den Menüpunkt **Inhalte**.
- Wählen Sie den Untermenüpunkt ***Neu***.
- Klicken Sie das Symbol zum Erzeugen eines neuen Produktes an.
- Geben Sie die Eigenschaften wie folgt ein:

Feld	Eingabe
Produkt	Messer-Set
Untertitel	5-teiliges Messer-Set aus Titan-Stahl für Gemüse, Fleisch und Knochen!
Beschreibung	**Ein Messer-Set der Superlative.** Für jede Anwendung in Ihrem Haushalt! Einschließlich Mehrzweckschere! **30 Jahre Zufriedenheitsgarantie**
Preis (DM)	198,00
Preis (EURO)	99,50

Tabelle 3.27: Produktdaten

- Klicken Sie den Reiter *Web* an.
- Geben Sie einen aktuellen Zeitraum für die Veröffentlichung ein.
- Klicken Sie auf die Schaltfläche SPEICHERN, um das Produkt zu speichern und das weitere Vorgehen festzulegen.
- Wählen Sie im Workflow *04 - Veröffentlichen im Web.*
- Klicken Sie auf die Schaltfläche SPEICHERN, um so das neue Produkt zu veröffentlichen.

3.10.4 Ergebnis

Wenn Sie die Kategorie *Küche* anwählen, sehen Sie, daß Ihr neues Produkt jetzt dieser Kategorie zugeordnet ist. Durch Anklicken des zugeordneten Inhaltssymbols können Sie sich das Produkt in der Gesamtansicht anzeigen lassen.

Abbildung 3.3: Listenansicht des neuen Produkts

3.10.5 Übung

- Legen Sie als ein weiteres Produkt *Flyer für Messer-Set* an.
- Ordnen Sie das neue Produkt der Kategorie *Werbeartikel* zu.
- Prüfen Sie, für welche Benutzer das Produkt sichtbar ist.

3.11 Erweiterte Aktionen: Hintergrundaktionen für die Bestellung

Ziel Eine eigene Aktion erstellen.

Bedeutung von Hintergrundaktionen verstehen lernen.

3.11.1 Beschreibung

Bestimmte Aktionen sollen zwar ausgeführt werden, es erfolgt allerdings keine Ausgabe auf dem Bildschirm. Diese Aktionen sind mit dem Attribut `BackgroundCall` zu versehen. Im CustomTag der Hintergrundaktion, muß festgelegt werden, was nach dessen Ausführung geschehen soll.

3.11.2 Beispiel

Für SiteMember sollen Bestellungen per Mausklick möglich sein. Der SiteMember soll auf ein Symbol in der Toolbar des jeweiligen Produktes klicken können, um dieses Produkt in seinen Warenkorb (`tutShoppingCart`) aufzunehmen.

3.11.3 Vorgehensweise

Schreiben Sie das CustomTag `[c:/gatebuilder/]customtags/tutorial/tutshoppingcartitemadd.cfm`, welches, falls noch kein Warenkorb vorhanden ist, einen neuen erstellt und das ausgewählte Produkt (`tutgbProduct`) in den Warenkorb aufnimmt.

Der Warenkorb wird mittels des Objektes `tutShoppingCart` dargestellt.

```
<cfsetting enablecfoutputonly=yes>
|Description: Add an object to a shopping cart of a user. If there is no actual
shopping cart for this user, create a new one. If the product to add is already
in the shopping cart increase the quantity.
|Remarks: Normally used as action.
|Usage: <tutshoppingcartitemadd
            DataSource="..."
            CartID="..."
            Quantity="...">
|Attributes: &CartID
|I/O: I
|R/O: O
|Type: string
|Defaults: Session.GateBuilder.CartID
|UIC-Variable:
|Description: ObjectID of the actual Shopping Cart
&Quantity
|I/O: I
|R/O: O
|Type: string
|Defaults:
|UIC-Variable:
|Description: Quantity of Products to buy
&DataSource
|I/O: I
|R/O: O
|Type: string
|Defaults: #Application.Gatebuilder.DataSource#
|UIC-Variable:
|Description: DataSource to use

<!- -- conceptware ag ---------------------------------------->
<cfparam name="Attributes.Quantity" default="1">
<cfparam name="Attributes.CartID" default="">
<cfparam name="Attributes.DataSource" default="#Application.Gatebuilder. ↩
   DataSource#">
<!-- Get ObjectID as Attribute from UIController -->
<cf_gbuicontrollerattributeget name="ObjectID" default="">
<cfif (len(trim(Attributes.CartID)) EQ 0) and not isDefined("Session. ↩
```

```
    Gatebuilder.CartID")>
    <!-- Create new Cart -->
    <cfa_contentobjectcreate
       datasource="#Attributes.Datasource#"
       typeid="#Application. ↩
          Gatebuilder.tutShoppingCartTypeID#"
       r_id="CartID">
    <cfa_contentobjectdata
       datasource="#Attributes.DataSource#"
       objectid="#CartID#">
       <cfa_contentobjectproperty name="UserName" value="#Session.GateBuilder.↩
          Security.Username#">
       <cfa_contentobjectproperty name="stItems" value="#StructNew()#">
    </cfa_contentobjectdata>
<cfelse>
    <cfif (len(trim(Attributes.CartID)) EQ 0) >
       <cfset CartID=Session.Gatebuilder.CartID>
    <cfelse>
       <cfset CartID=Attributes.CartID>
    </cfif>
</cfif>
<!-- Get the Cart object -->
<cfa_contentobjectget datasource="#Attributes.DataSource#" objectid="#CartID#"↩
       r_stobject="stCart">
<!-- Set the Quantity -->
<cfif len(trim(Attributes.Quantity)) GT 0>
    <cfset stItems = stCart.stItems>
    <cfif StructKeyExists(stItems,"#Attributes.ObjectID#") >
       <!-- Quantity is already there, so add one more to the shopping cart -->
       <cfset stItems[ObjectID] = stItems[ObjectID] + 1>
    <cfelse>
       <!-- The first product in this shopping Cart -->
       <cfset stItems[ObjectID] = 1>
    </cfif>
<cfelse>
    <cfset stItems[ObjectID] = "#Attributes.Quantity#">
</cfif>
<!-- Change the Content Object -->
<cfa_contentobjectdata datasource="#Attributes.DataSource#" objectid="#CartID#">
    <cfa_contentobjectproperty name="Label" value="#Session.GateBuilder ↩
       .Security.Username#">
    <cfa_contentobjectproperty name="stItems" value="#stItems#">
</cfa_contentobjectdata>
<!-- Store CartID in Session -->
<cfset Session.Gatebuilder.CartID = CartID>
<!-- Get new URL Address from UI-Controller and switch to this address -->
<cf_gbuicontrollerbaseurlget r_baseurl="BaseURL">
<cflocation url="#BaseURL#">

<cfsetting enablecfoutputonly="no">
```

Tabelle 3.28: Code-Beispiel: tutShoppingCartItemAdd

3.11.4 Übung

Feld	Eingabe
ObjectID	62944FAE-C992-11D3-A41000500D05FA4
Label	`tutShoppingCartItemAdd`
CustomTag	`tutShoppingCartItemAdd`
Parameter(s)	
BackgroundCall	(Feld aktivieren)
ConfirmCall	
SecureCall	
Beschreibung	Insert product into shopping cart.
Zugänglich für	SiteMembers
Cache aktualisieren	(Feld aktivieren)
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabe abbrechen
HILFE	Online-Hilfe aufrufen

Tabelle 3.29: Die Aktion tutShoppingCartItemAdd

Erstellen Sie die Aktion `tutShoppingCartItemAdd` (vgl. Abschnitt 3.8 „Die Aktionen“ ab S.31) für SiteMembers. Geben Sie bei der Einrichtung der Aktion an, dass diese Aktion als *BackgroundCall* ohne *ConfirmCall* (Sicherheitsabfrage) gestartet werden soll.

3.12 Toolbars

Ziel Toolbars verwalten.

Toolbars in die Anwendung integrieren.

3.12.1 Beschreibung

Toolbars sind Relationen vom Typ Menü (`gbMenu`). Um für eine Anwendung Toolbars zu definieren, werden diese unter einer Root (`gbRelation`) mit verschiedenen, eindeutigen Namen erstellt. Die Knoten der ersten Ebene des Relationsbaums entsprechen den einzelnen Toolbars. Beim Aufruf wird die Toolbar über den Namen des zugehörigen Knotens eindeutig identifiziert.

Der Aufbau einer Toolbar ist analog zum Menü, mit der Besonderheit, daß die eigentliche Toolbar erst auf der 2. Ebene beginnt.

Dies können Sie am CustomTag `[c:\gatebuilder\]customtags\general\ui\gbtoolbar.cfm` ebenfalls nachvollziehen. Nach der Zusammenstellung der Parameter und Auflösung des Namens in die NodeID des Knotens wird das CustomTag `gbMenu` zur Darstellung der Toolbar aufgerufen.

3.12.2 Beispiel

Für das Beispiel muß die *Default_Toolbar* aus der Relation `tut_gbToolbarDE`, die von GateBuilder bei der Anzeige eines Content-Objekts verwendet wird, um eine zusätzliche Funktion für SiteMembers (In Warenkorb stellen - `tutShoppingCartItemAdd`) ergänzt werden (vgl. 3.9 „Relationen für Menü und Kategorienbaum“ auf S.33).

3.12.3 Vorgehensweise

Als Übung können Sie nun die Toolbar ergänzen. Vorab sollten Sie jedoch nochmals die dem neuen Inhalt momentan zugeordnete Toolbar prüfen. Vergleichen Sie bitte die Toolbar (DEFAULT_TOOLBAR) für Seitenbesucher ohne Anmeldung (WebUser) mit der für die Gruppe SiteMembers, um spätere Änderungen nachvollziehen zu können.

3.12.4 Übung

Melden Sie sich als Administrator an und erweitern Sie die Toolbar DEFAULT_TOOLBAR um folgende neue Option:

Eigenschaft	Wert
Name	ItemAdd
Beschreibung	zum Warenkorb hinzufügen
Sortierreihenfolge	999
Zugänglich für	SiteMembers
Bild(er)	`/tutorial/tutorial/images/system/toolbar/shopping.gif`
Aktion	`TutShoppingCartItemAdd`
Fenstername / Aussehen	

Tabelle 3.30: Hinzufügen einer Toolbar-Schaltfläche

3.12.4.1 Ergebnis

Vergleichen Sie jetzt noch einmal die Toolbar für Gäste und SiteMembers.

Sie werden feststellen, daß die Toolbar jetzt über die neue Schaltfläche ZUM WARENKORB HINZUFÜGEN verfügt. Klicken Sie darauf, wird das angezeigte Produkt in den Warenkorb aufgenommen.

3.12.4.2 Übung

Erzeugen Sie das CustomTag `tutshoppingcart.cfm` und die zugehörige Aktion, die es ermöglicht, die ausgewählten Produkte und die gewünschte Menge anzuzeigen. Neben einer eMail an den Benutzer als Bestätigung soll die Produktbestellung außerdem per eMail an den Vertrieb (`tut_gbapplicationDE.eMail`) übermittelt werden.

Sollten Sie bei der Erstellung des angegebenen CustomTags Probleme haben, können Sie die Datei aus dem Verzeichnis mit den bereits erstellten CustomTags kopieren:

`[c:\gatebuilder]\customtags\tutorial\`

Andernfalls speichern Sie das CustomTag `tutshoppingcart.cfm` im Verzeichnis

`[c:\gatebuilder\]customtags\tutorial\`.

Die Aktion muß wie folgt angelegt werden:

Feld	Eingabe
ObjectID	62944FAE-C992-11D3-A41000500D05FA4
Label	`tutShoppingCart`
CustomTag	`tutShoppingCart`
Parameter(s)	
BackgroundCall	(Feld aktivieren)
ConfirmCall	
SecureCall	
Beschreibung	Display shopping cart.
Zugänglich für	SiteMembers
Cache aktualisieren	(Feld aktivieren)
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabe abbrechen
HILFE	Online-Hilfe aufrufen

Tabelle 3.31: Die Aktion tutShoppingCart

Die Menüstruktur sollte wie folgt geändert werden, soweit nicht bereits geschehen:

Hauptmenü	Untermenü	Rechte	Erläuterung
MeinShop		Zugriff: SiteMembers	Benennen Sie den Menüpunkt **Mein Gate** um.
	MeinShop	Zugriff: SiteMembers	Benennen Sie den Untermenüpunkt ***Mein Gate*** um.
	Kennwort	Zugriff: SiteMembers	
!	Bestellung	Zugriff: SiteMembers	Aktion: `tutshoppingcart`

Tabelle 3.32: Übung zur Änderung des Menüs

3.13 Design

3.13.1 Cascading Style Sheets

Ziel Cascading Style Sheets verwenden.

Cascading Style Sheets für die Anwendung anpassen.

3.13.1.1 Beschreibung

GateBuilder verwendet Cascading Style Sheets zur Formatierung der HTML-Seiten.

Um der Funktionalität der verschiedenen Browser gerecht zu werden, müssen für einzelne Browser unterschiedliche Cascading Style Sheets zur Verfügung gestellt werden. Unterschieden wird zwischen dem Microsoft Internet Explorer und anderen Browsern.

Die einzelnen Anzeigefunktionen (Controls) besitzen eigene Class-Definitionen, so dass durch Anpassung der jeweiligen Klasse das Design Ihrer Anwendung verändert werden kann.

Das CustomTag `gbFormatting` (siehe auch Abschnitt 3.4 „Aufbau der WebSite“ auf S.12), der aus einem öffnenden und einem schließenden Teil besteht, übernimmt das Schreiben der HTML-Header und der Verknüpfungen mit Ihren Cascading Style Sheets (`gbCSSDirect`) und Java-Skripts.

Die Einstellungen des Cascading Style Sheets liest das Custom Tag aus dem `gbCSSDirect`-Objekt, das in der Anwendung (`tut_gbApplicationDE`) definiert wird.

Wurden Änderungen am `gbCSSDirect`-Objekt vorgenommen, wird dem Web-Server auf dieser Grundlage eine neue CSS-Datei zur Verfügung gestellt.

Das End-Tag von `gbFormatting` schreibt den HTML-Footer.

3.13.1.2 Beispiel

Betrachten Sie nun die HTML-Seite im Web-Browser. Es liegt jetzt eine fertige Anwendung vor, bei der jedoch noch das Design des Menüs und der Produktgruppen angepasst werden sollte.

3.13.1.3 Vorgehensweise

- Melden Sie sich als Benutzer *Designer* an.
- Klicken Sie auf **Admin**.
- Wählen Sie den Menüpunkt ***Konfiguration***.
- Klicken Sie auf den Untermenüpunkt ***Styles***.
- Wählen Sie das Objekt `tut_gbCCSDirect` aus.
- Klicken Sie die Schaltfläche BEARBEITEN des Objektes.
- Nehmen Sie folgende Änderungen vor:

Feld	Eingabe
Label	tut_gbCSSDirect
Beschreibung	Cascading Style Sheet for Tutorial (IE)
StyleSheet	BODY { background-color : #FFFFFF; margin : 0; margin-left : 0; margin-right : 0; margin-bottom : 0; border-style : none; font-family : Arial; } TABLE { } TR { } TH { } TD { } SELECT { } INPUT { } INPUT.TEXTFIELD { width : 100%; } DESCRIPTION { } A { font-size : x-small; }

Tabelle 3.33: Bearbeiten eines Cascading Style Sheets

Hauptmenü	**Untermenü**

.gbmainmenu
{
color : #111183;
text-decoration : none;
padding : 5;
font-weight : bold;
font-size : small;
}
 	
.gbmainmenuselected
{
color : #7070B5;
text-decoration : none;
padding : 5;
font-weight : bold;
font-size : x-small;
}
	.gbsubmenu
	{
	color : #111183;
	text-decoration : none;
	padding : 2;
	font-weight : bold;
	font-size : x-small;
	}
	
	.gbsubmenuselected
	{
	color : #7070B5;
	text-decoration : none;
	padding : 2;
	font-weight : bold;
	font-size : x-small;
	}

Kategorien und Unterkategorien	**Kategorien und Unterkategorien**

.gbcategorytoplevel
{
color : #111183;
text-decoration :none;
font-weight : bold;
font-size : medium;
}
 	
.gbcategorytoplevelselected
{
color : #7070B5;
text-decoration : none;
font-weight : bold;
font-size : medium;
}
	.gbcategorysublevel
	{
	color : #111183;
	text-decoration : none;
	font-weight : bold;
	font-size : small;
	}
	
	.gbcategorysublevelselected
	{
	color : #7070B5;
	text-decoration : none;
	font-weight : bold;
	font-size : small;
	}

Tabelle 3.34: Einstellungen für Cascading Style Sheet

- Klicken Sie nach den Änderungen auf die Schaltfläche SPEICHERN, um das neue Cascading Style Sheet zu sichern.

3.13.1.4 Übung

- Enfernen Sie den Eintrag für die Hintergrundfarbe der Kategorienübersicht im CustomTag `tutCategoryOverviewAndUserConsole`.

3.13.2 Die Ressourcen

Ziel Ressourcen in die Anwendung integrieren.

3.13.2.1 Beschreibung

Ressourcen sind sprachabhängige Textbausteine, die auf der WebSite als Meldungen, Labels oder zur Darstellung einfacher Beschreibungstexte verwendet werden.

Die Ressourcen einer Sprache sind durch folgende Parameter eindeutig definiert:

Attribut	Beschreibung
Language	Sprache, in welcher der Textbaustein angezeigt werden soll.
Owner	Besitzer der Ressource, welcher bei der Initialisierung der Anwendung für diese festgelegt wird (CustomTag `gbApplicationInitialize`) (vgl. hierzu auch Abschnitt 3.7 „Anpassung der application.cfm“ auf S.30)
Resource_ID	eindeutige Nummer des zu verwendenden Textbausteins

Tabelle 3.35: Parameter einer Ressource

Kodieren Sie Texte in Ihren CustomTags nicht hart. Sie können dann jederzeit Ihre Anwendung um weitere Sprachen erweitern, da nur die entsprechenden Textbausteine einzupflegen sind. Auch für den Pfleger der WebSite (angemeldet als Designer), ist es wesentlich einfacher den jeweiligen Textbaustein direkt auf der aktuellen Seite im Browser zu ändern, als das dazugehörige CustomTag zu suchen und dort den Text zu ändern.

Ressourcen können sowohl vom Administrator als auch vom Designer gepflegt werden, wobei der Designer den Vorteil des *In Context Editing* (Bearbeiten im aktuellen Anwendungskontext) hat.

3.13.2.2 Beispiel

Als Standardbesitzer der Ressourcen wird `gb` verwendet. Wollen Sie Ressourcen für Ihre Anwendung nutzen, geben Sie als Owner `tut` an.

Die Beschriftung des Warenkorbs (CustomTag `tutshoppingcart.cfm`) soll in der GateBuilder-Anwendung der Multipart AG ebenfalls über Ressourcen ausgegeben werden.

3.13.2.3 Vorgehensweise

Um den Textbaustein in das CustomTag einzubauen, müssen Sie wie folgt vorgehen:

- Melden Sie sich als Administrator an.
- Wählen Sie das Menü **Admin** aus.
- Klicken Sie auf den Menüpunkt ***Konfiguration***.
- Wählen Sie den Untermenüpunkt ***Ressourcen***.
- Klicken Sie auf die Schaltfläche `NEU`, um eine neue Ressource anzulegen.
- Erstellen Sie folgende Ressource:

Feld	Eintrag
Owner	TUT
Kategorie	StaticText
Englisch	Shopping Cart
Deutsch	Warenkorb

Tabelle 3.36: Erstellen einer Ressource

- Klicken Sie auf die Schaltfläche SPEICHERN, um die Änderungen zu übernehmen.
- Merken Sie sich die angezeigte ID der Ressource, da diese nun in das CustomTag eingebaut werden muß.
- Öffnen Sie das CustomTag `tutshoppingcart.cfm`, das sich im Verzeichnis `[c:\gatebuilder\]customtags\tutorial` befindet.
- Ersetzen Sie den Text *Ihr Warenkorb* in Ihrem CustomTag durch folgenden CustomTag-Aufruf:

```
...
<tr><td colspan="2">

<h3>
<cf_gbgetresource
       Resource_ID="ID Ihrer neuen Resource"
       owner="tut">
</h3>
...
```

Tabelle 3.37: Code-Beispiel: Verwendung einer Ressource

- Speichern Sie Ihre Änderungen.

3.13.2.4 Ergebnis

Sie haben eine neue Ressource in Ihre Anwendung eingebunden, welche Sie als Benutzer *Designer* auf unproblematische Art und Weise verändern können. Gehen Sie hierzu wie folgt vor:

- Melden Sie sich als Benutzer Designer an.
- Wählen Sie die Kategorie *Küche* aus.
- Klicken Sie auf die Schaltfläche ZUM WARENKORB HINZUFÜGEN des Produktes *Messer-Set*, um es in Ihren Warenkorb zu übernehmen.
- Wählen Sie den Menüpunkt **Mein Shop**.
- Klicken Sie auf den Untermenüpunkt ***Bestellung***.

An der Überschrift *Warenkorb* erscheint nun das Symbol für Ressource.

- Klicken Sie dieses Symbol.
- Ändern Sie die Ressource von *Warenkorb* in *Mein Warenkorb* um.
- Klicken Sie auf die Schaltfläche SPEICHERN, um die Änderungen zu übernehmen.
- Klicken Sie auf die Schaltfläche SCHLIESSEN, um den Ressourcen-Editor zu beenden.
- Aktualisieren Sie Ihre HTML-Seite, um die Ressourcen-Änderungen zu sehen.

3.13.3 Informationstexte anpassen

Ziel Informationstexte (`gbInformation`-Objekte) in die Anwendung intergrieren.

Informationstexte pflegen.

3.13.3.1 Beschreibung

`gbInformation`-Objekte werden zur Darstellung größerer formatierter Texte in der Anwendung verwendet, z.B. für Kontaktinformationen, Wegbeschreibungen, etc.

Es gibt zwei `gbInformation`-Objekte, die standardmäßig von GateBuilder verwendet werden: die Kontaktinformation und die Registrierungsinformation.

Diese Objekte werden im `gbApplication`-Objekt für die gesamte Anwendung definiert.

3.13.3.2 Beispiel

In dem Beispiel wird eine spezielle Registrierungsinformation benötigt, die dem Benutzer mitteilt, dass er seine Anmeldung per Fax einschließlich des Gewerbescheins durchführen soll.

Klickt der Benutzer jetzt in der UserConsole auf den Link *Registrieren*, erhält er diese neue Registrierungsinformation als Dialog, den er danach direkt wieder schließen kann.

3.13.3.3 Vorgehensweise

Um eine Information (z.B. Registrierung) zu bearbeiten, gehen Sie wie folgt vor:

- Melden Sie sich als Administrator an.
- Klicken Sie auf **Admin**.
- Klicken Sie auf den Menüpunkt ***Konfiguration***.
- Wählen Sie den Untermenüpunkt ***Information***.
- Wählen Sie das Objekt `tut_gbRegistration (DE)` aus.
- Klicken Sie auf die Schaltfläche BEARBEITEN.
- Ändern Sie die Eigenschaft Information wie folgt ab: (siehe Tabelle 3.38).

Herzlichen Dank für Ihr Interesse an unseren Produkten. Einzelhändlern geben wir die Web-Anwendung mit Preisanzeige und Bestellfunktion frei. Senden Sie für eine Freigabe Ihren Gewerbenachweis an folgende FAX-Nummer: **0123-456789**. Mit freundlichen Grüßen Ihr Multipart-Team

Tabelle 3.38: Registrierungsinformation

- Klicken Sie auf die Schaltfläche SPEICHERN, um die Änderungen zu übernehmen.

Überprüfen Sie, ob für das `tut_gbApplicationDE`-Objekt `tut_gbRegistration` als `RegistrationInfo`-Objekt eingetragen ist.

Gehen Sie hierzu wie bei der Änderung der Registrierungsparameter vor (siehe hierzu auch Abschnitt 3.5.1 „Die Anwendungsumgebung“ auf S.17).

Das Verhalten des Registrierungsdialoges läßt sich im Session-Objekt unter `RegistrationAskUser` und `RegistrationAskUserData` ändern (siehe dazu Abschnitt 3.10 auf S.18).

3.13.3.4 Ergebnis

Wenn Sie auf Ihrer Homepage jetzt den Link zur Registrierung anklicken, wird Ihnen der neue Informationstext angezeigt.

3.13.4 Typ-Spezifikation

Ziel Layout-Spezifikationen (`gbTypeSpec`-Objekte) einrichten. Layout-Spezifikationen auf Content-Typen anwenden.

3.13.4.1 Beschreibung

Die Typ-Spezifikation (`gbTypeSpec`) gibt Ihnen die Möglichkeit, in Abhängigkeit vom Content-Typ Einstellungen für das Layout zu definieren. Für jeden Objekt-Typ, der in Ihrer Anwendung verwendet wird, müssen Sie ein beschreibendes `gbTypeSpec`-Objekt anlegen. In diesem Objekt können Sie das Layout des Content-Typs (`SpecTypeID`) festlegen, d.h. beispielsweise welche Images und welche Überschriften speziell für diesen Typ verwendet werden sollen.

Wurde die Typ-Spezifikation bereits angelegt, kann sie der Benutzer *Designer* direkt im Anwendungskontext pflegen.

3.13.4.2 Beispiel

In unserem Beispiel wurde bereits beim Erzeugen des neuen GateTypes `tutProducts` (`tutgbProduct`) ein `gbTypeSpec`-Objekt für diesen GateType angelegt (siehe 3.6.2 „Duplizieren eines vorhandenen GateTypes“ auf S.23).

Dieses Objekt entspricht dem `gbTypeSpec`-Objekt des Referenz-GateTypes `gbProduct`, für die Multipart AG soll das Produkt (`tutgbProduct`) allerdings neue Bezeichnungen erhalten.

3.13.4.3 Vorgehensweise

- Melden Sie sich als Benutzer *Designer* an.
- Klicken Sie auf den Menüpunkt **Admin**.
- Wählen Sie den Menüpunkt ***Konfiguration***.
- Klicken Sie auf den Untermenüpunkt ***Types***.
- Suchen Sie den Type `tut_gbProduct (DE)`.
- Klicken Sie auf die Schaltfläche BEARBEITEN.
- Nehmen Sie folgende Änderungen vor:

Feld	Eintrag
Label	Tutorial Product
Title	Tutorial Product
Description	Tutorial Products for Multipart Ltd.
MyGateTitle	My Tutorial Products
TopTitle	Top Tutorial Products
ImageURL	`images/system/objicons/icon_product.gif`
MyGateImageURL	`images/system/top/top_product.gif`
TopImageURL	`images/system/top/top_product.gif`
SpecTypeID	9E1F7A1A-6D18-11D4-A4A2005004F6359B
Sprache	Deutsch oder English
Cache aktualisieren	(aktivieren)
SPEICHERN	Eingaben speichern
ABBRECHEN	Eingabemaske beenden, ohne die Eingaben zu speichern
HILFE	Online-Hilfe aufrufen

Tabelle 3.39: Spezifikation des Layout ProductType

3.14 Ausblick

Sie haben nun eine komplette Anwendung mit GateBuilder erstellt.

Zur Kontrolle vergleichen Sie Ihre Homepage mit der Abbildung 3.4.

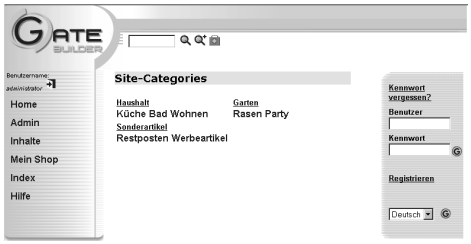

Abbildung 3.4: Die fertige Shopping-Anwendung der Multipart AG

Obwohl die hier vorgestellten Layout-Anpassungen und Aktionen noch recht einfach gehalten sind, können Sie mit einem vergleichsweise geringen Aufwand die Bestellungen mit Hilfe des Warenkorbs auch in Ihr Warenwirtschaftssystem übernehmen.

Als nächsten Schritt sollten Sie Ihre im Rahmen dieses Tutorials erworbenen Kenntnisse am besten mit Hilfe des Administratoren-Handbuchs vertiefen.

Viel Spaß und Erfolg bei der Arbeit mit GateBuilder.

Tabellenverzeichnis

1.1 Schreibweisen 4

2.1 Liste der Workflow Templates 5

3.1 Parameter zum Erstellen der Tutorial-Anwendung 10
3.2 Erläuterungen zun Abbildung 3.2 12
3.3 Das Template index.cfm 13
3.4 CustomTags für den linken Rand der Seite 14
3.5 Der linke Rand 14
3.6 CustomTags für den oberen Rand der Seite 15
3.7 Der obere Rand der Seite 16
3.8 CustomTags für den Hauptteil der Seite 16
3.9 Der Hauptteil 16
3.10 Einstellung der Registrierung 18
3.11 Konfiguration der Anwendung 19
3.12 Konfiguration der Benutzerumgebung mit gbSession-Objekten 21
3.13 Eigenschaften eines GateTypes 22
3.14 Benötigte Handler für GateTypes 22
3.15 Parameter zum Duplizieren des GateTypes Products 24
3.16 Erweiterung des Handlers create.cfm 26
3.17 Änderungen in der EditStep-Methode 27
3.18 Session.GateBuilder.Security 27
3.19 Erweiterung des Handlers printdisplay.cfm 28
3.20 Parameter für Content-Typ tutShoppingCart 29
3.21 Eigenschaft des Content-Typs tutShoppingCart 29
3.22 Code-Beispiel: Ergänzungen zu Datei application.cfm 31
3.23 Code-Beispiel: CustomTag tutCategoryOverviewAndUserconsole 32

3.24 Erstellen der Aktion tutCategoryOverviewAndUserConsole 32
3.25 Menü-Definition 33
3.26 Darstellung der Produktgruppen als Kategorienbaum 35
3.27 Produktdaten 36
3.28 Code-Beispiel: tutShoppingCartItemAdd 39
3.29 Die Aktion tutShoppingCartItemAdd 40
3.30 Hinzufügen einer Toolbar-Schaltfläche 41
3.31 Die Aktion tutShoppingCart 42
3.32 Übung zur Änderung des Menüs 42
3.33 Bearbeiten eines Cascading Style Sheets 43
3.34 Einstellungen für Cascading Style Sheet 44
3.35 Parameter einer Ressource 45
3.36 Erstellen einer Ressource 46
3.37 Code-Beispiel: Verwendung einer Ressource 46
3.38 Registrierungsinformation 48
3.39 Spezifikation des Layout ProductType 49

Abbildungsverzeichnis

3.1 Struktur der Objekte 9
3.2 Layout der Tutorial-Site (Template index.cfm) 12
3.3 Listenansicht des neuen Produkts 37
3.4 Die fertige Shopping-Anwendung der Multipart AG 50

Index

A

A (Anker-Tag) 43
Absolute path for handler root 24
Activate 22
Administratoren 2, 3
Administrators 2, 3
Aktion 8, 31
Aktion einbinden 8
Aktion umsetzen 8
Allaire Spectra 1
Anwedungsumgebung anwenden 8
Anzeige-Methode 27
Application Name 10
Application Path 10
Application Security Context 10
Application-Scope 17
application.cfm 10, 30
AppRootUrl 19
AppServerMapping 19
Architektur 7
Archive 22
ATTR_ACTIVE 5
ATTR_ARCHIVED 5
Autoren 2

B

BackgroundCall 32, 37, 40, 42
Banner 23
BannerGate 1
Baumstruktur 33
Benutzer 2
Benutzergruppen 2, 8
Besucher 2
bNewsletterSubscribed 21
BODY 43
Body 22
Browser 42

C

Cache 19, 28, 32, 40, 42, 49
Cache aktualisieren 20
CalendarGate 1
Cascading Style Sheets 9, 11, 21, 42
cf_gbgate 16
cf_gblogout 14
cf_gbmenu 14, 15
cf_gbsearch 15
CFML 1
Class-Definitionen 42
Clone-Skript 12
clonegatetype.cfm 23, 24
ColdFusion Markup Langugage 1
ColdFusion-Server 19
ConfirmCall 40, 42
ContactGate 1
ContactInfo 19
Content-Objekt 40
Content-Typ 8, 24, 28, 48
Content-Typen erzeugen 8
Content-Typen unter Spectra anlegen 28
Contributors 2
ControlEvent 27
Controls 42
Create 22
create.cfm 26
CSSID 20, 21
CSSID_ALT 21
CSSID_Alt 20
CustomTag 1, 11, 14, 32, 40, 42

D

DEFAULT_TOOLBAR 41
Delete 22
DESCRIPTION 43
Description 49
Design 42
Design anpassen 9
Designer 2, 48
Designers 2
Destination DataSource 24
Display 22
Display-Handler 23
DocumentGate 1
Dokumente 23
Duplizieren 17

E

eBusiness Application Suite 1
Edit 22
edit.cfm 26

Editors 2
EditStep 26
editstep.cfm 26
Eigene Anwendung erzeugen 9
Eigene Workflows 6
eMail2FriendInfo 19
Entwickler 2
Erweiterte Aktion 37
Erweiterung durch eigene Workflows 6
Execute-Methode 31
Extranet 1

F

Framework 1
Workflow 5–6
Fulldisplay 22
Fullname 27

G

GateBuilder-Anwendung erzeugen 8
GateType 22, 23, 26, 28, 49
GateTypes 8–10, 20–24, 26
gbAction 9, 31
gbApplication 9, 11, 17, 18, 20, 30, 47
gbApplicationDE 10
gbApplicationInitialize 30, 45
gbBanner 9
gbCalendar 9
gbCategoryOverview 35, 36
gbCategoryRootDE 18
gbContact 9
gbCSSDirect 9, 11, 21, 43
gbCSSDirectALT 21
gbDocument 9
gbFormatting 13, 43
gbGate 34
gbgateinvoker.cfm 30
gbGateType 23
gbHelpContextDE 18
gbHelpContextRootDE 18
gbHelpRootID 18
gbInformation 9, 47
gbKeywords 19
gbLink 9
gbListControl 22
gbMenu 40
gbMenuRootID 18
gbNavItem 9, 33, 34
gbNavItems 31
gbNews 9
gbProduct 9, 23
gbPublish 26
gbRelation 9, 40
gbRelations 33
gbSession 9, 11, 17, 20, 28
gbSessionDE 11
GBSESSIONID 19
gbSessionID 20
gbToolbarRootDE 18
gbTypeSpec 9, 23, 48
gbWorkflow_197 5
gbWorkflow_198 5
gbWorkflow_199 5
gbWorkflow_200 5
gbWorkflow_237 5
gbWorkflowExec 18
gbWorkflowExecRootID 18
gbWorkflowExecute 5
gbWorkflowInit 18
gbWorkflowInitiate 5
gbWorkflowInitRootID 18
Grundeinstellungen 17
Grundfunktionen 7

H

Handler 26
Handler Root 29
Hauptteil 16
Hintergrundaktion 37
HTML-Footer 43

I

ImageURL 49
In Context Editing 45
Include-Datei 13, 16
index.cfm 10, 12, 13, 16, 30
InfoGate 1
Informationsobjekte 17
Informationstext 47
Inhalte 36
Inhalte veröffentlichen 36
INPUT 43
INPUT.TEXTFIELD 43
Intranet 1
ItemAdd 41

J

Java-Skript 43

K

Kategorie 11, 34, 46
Kategorienbaum 34
kategorienbaum 33
Kennwort 27
KEYWORDCATEGORYID 19
Kontaktadressen 23

L

Label 18, 29, 32, 40, 42, 43, 49
LANDSCAPE-Mode 15
Language 20, 21, 45
LastNotifyDateTime 21

Laufwerk-Mapping 17
Layout 16
Layout festlegen 8
lcommentedTypeIDs 21
_left.cfm 10, 13, 14, 16
lfavoriteObjectIDs 21
lgbApplicationIDs 30
lGroups 27
LinkGate 1
Links 23
ListDisplay 22
lNewsletterCategories 21
Login-Felder 31
LoginIsMandatory 19
LoginIsRegistration 19
lUserNotifyCategories 21
lUserNotifyGateType 21
lUserSiteCategories 21
lUserSiteGateType 21
lUserSiteGateTypes 20

M

_main.cfm 10, 13, 16
Mappings 19
Menü 11, 33
Menüstruktur 33
Menüstruktur abbilden 8
MENUHOME 19
MyGateImageURL 49
MyGateTitle 49

N

News 23
Newsletter 23

O

ObjectID 18, 28
Objekt-Typ 48
ObjektIDs 10
Online-Hilfe 19, 21, 32, 40, 42, 49
Owner 10, 45, 46

P

Password 27
Paths 19
Pfade 17
PORTRAIT-Mode 14
Prefix 10
Prefix for new GateType 24
PrintDisplay 22, 27
PrintDisplay-Handler 28
ProductGate 1
Produkte 23
ProgPath 19
PublishInfo 22

R

Rechte 8
Rechte-Relationen 9
Redakteure 2
Reference GateType 24
RegistrationAskAccount 18, 19
RegistrationAskUser 48
RegistrationAskUserData 18, 19, 48
RegistrationDirectAccess 19
RegistrationFields 19
RegistrationInfo 19, 48
RegistrationShowInfo 18, 19
Registrieren 47
Registrierungslink 31
Relationen 11, 33, 34
Relative path for handler root 24
Resource_ID 5, 45
Resources 9
Ressourcen 45
Root-Objekt 33

S

Save 22
Schnellsuche 15
SecureCall 40, 42
SELECT 43
Session-Scope 17, 27
session.gatebuilder.language 17
Session.GateBuilder.Security 27
SessionScope 28
Sicherheitsabfrage 40
SiteMember 7, 8, 17, 28, 40
SiteMembers 2, 24
Sortierreihenfolge 41
Source DataSource 24
Source DSN 10
Spectra 25
SpecTypeID 48, 49
sprachabhängig 9
Sprachauswahl 31
Sprache 18, 20, 21, 49
Standardsprache 20
Standardsprachen 16
StaticText 46
stItems 29
Structure 29
StyleSheet 43
SubTitle 22
Suchmaske 15
System Entities 29

T

TABLE 43
Target DSN 10
TaskDisplay 22
TD 43

Teaser 22
Template 10
Termine 23
TH 43
Title 22, 49
Toolbar 40
Toolbars 11
TopDisplay 22
TopImageURL 49
TopTitle 49
TR 43
tut_gbApplicationDE 10, 11, 18, 20, 30
tut_gbapplicationDE.eMail 41
tut_gbCategoryDE 18
tut_gbCSSDirect 11, 20, 43
tut_gbCSSDirectALT 11
tut_gbCSSDirectAlt 20
tut_gbFooter (DE) 19
tut_gbHelpDE 18
tut_gbMenuID 18
tut_gbProduct 26
tut_gbRegistration 18
tut_gbRegistration (DE) 19
tut_gbSessionDE 11, 19–21
tut_gbToolbarDE 18, 40
tutCategoryOverviewAndUserConsole . . . 32, 36, 44
tutcategoryoverviewanduserconsole 33
tutcategoryoverviewanduserconsole.cfm 32
tutgbCategoryOverviewAndUserConsole 33
tutgbProduct 28, 30, 36, 38, 49
tutProducts 26, 49
tutShoppingCart 28–30, 38, 42
tutshoppingcart 42
tutshoppingcart.cfm 41, 45, 46
tutShoppingCartItemAdd 32, 40
Typ-Spezifikation 48
TypeID 30

U

Untermenü 15
UploadPath 19
_upper.cfm 10, 13, 15, 16
URL 10
UserConsole 19, 31, 47
USERLOG 19
Username 27, 29
UserNotifyFrequencies 21
UserNotifyMethod 21
UserSiteTimeSpan 21

V

Validate 22
Vollanzeige 22

W

Warenkorb 8
Web-Browser 43
Web-Portal 1
WebPath 19
WebRoot 10
WebSite-Besucher 2
WebUser 41
WebUsers 2
Workflow
    Aufbau 5
    Eigener 6
Workflow Framework 5–6
Workflow Templates 5

Z

Ziel des Tutorials 7
Zugriffsrechte 33, 34